UNCLASSIFIED

AD NUMBER

ADB244332

NEW LIMITATION CHANGE

TO

Approved for public release, distribution unlimited

FROM

Distribution authorized to U.S. Gov't. agencies only; Proprietary Info; Dec 98 Other requests shall be referred to USAMRMC, Ft. Detrick, MD 21702-5012

AUTHORITY

USAMRMC ltr dtd 4 Jan 2000

THIS PAGE IS UNCLASSIFIED

AD_________________

AWARD NUMBER DAMD17-94-J-4500

TITLE: Dual-Use Telemedicine Support System for Pathology

PRINCIPAL INVESTIGATOR: Diane Conti

CONTRACTING ORGANIZATION: Kensal Corporation
Tucson, Arizona 85712

REPORT DATE: December 1998

TYPE OF REPORT: Final

PREPARED FOR: U.S. Army Medical Research and Materiel Command
Fort Detrick, Maryland 21702-5012

DISTRIBUTION STATEMENT: Distribution authorized to U.S. Government agencies only (proprietary information, Dec 98). Other requests for this document shall be referred to U.S. Army Medical Research and Materiel Command, 504 Scott Street, Fort Detrick, Maryland 21702-5012.

REPORT DOCUMENTATION PAGE

OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)

2. REPORT DATE
December 1998

3. REPORT TYPE AND DATES COVERED
Final (1 Oct 94 - 30 Sep 98)

4. TITLE AND SUBTITLE
Dual-Use Telemedicine Support System for Pathology

5. FUNDING NUMBERS
DAMD17-94-J-4500

6. AUTHOR(S)
Diane Conti

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Kensal Corporation
Tucson, Arizona 85712

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
US Army Medical Research & Materiel Command
Fort Detrick MD 21702-5012

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

19990526 100

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Distribution authorized to U.S. Government agencies only (proprietary information, Dec 98). Other requests for this document shall be referred to U.S. Army Medical Research and Materiel Command, 504 Scott Street, Fort Detrick, Maryland 21702-5012.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

Traditional pathology is practiced utilizing a microscope for examination of glass slides containing tissue samples. Being able to see only a minuscule portion of the entire coverslip through the microscope, the pathologist must search for any area of interest by perusing the slide at a low magnification. Although it is easy to switch to a higher magnification as desired, one of the difficulties arises when attempting to return to a previous area of interest. Attempts to achieve telepathology utilizing this technology are primitive. Kensal Corporation has developed a revolutionary tool for pathologists employing digital technology which eliminates the inherent limitations of traditional microscopy. "Lensless microscopy" enables a pathologist to capture and display the entire coverslip and view it on a high resolution monitor. This image is the scout image. Areas of interest can be readily identified and selected for viewing at higher magnification. Since the coordinates of each new magnification are automatically recorded and saved, the pathologist can return to the precise location of any area of interest previously viewed. Because it is now in digital form, the entire coverslip can be archived as a case file and transmitted for remote consultations. True telepathology now exists.

14. SUBJECT TERMS
telepathology, lensless microscopy, imaging system

15. NUMBER OF PAGES
538

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	Limited

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102

DTIC QUALITY INSPECTED 4

FOREWORD

Opinions, interpretations, conclusions and recommendations are those of the author and are not necessarily endorsed by the U.S. Army.

( ) Where copyrighted material is quoted, permission has been obtained to use such material.

( ) Where material from documents designated for limited distribution is quoted, permission has been obtained to use the material.

DC Citations of commercial organizations and trade names in this report do not constitute an official Department of the Army endorsement or approval of the products or services of these organizations.

( ) In conducting research using animals, the investigator(s) adhered to the "Guide for the Care and Use of Laboratory Animals," prepared by the Committee on Care and Use of Laboratory Animals of the Institute of Laboratory Animal Resources, National Research Council (NIH Publication No. 86-23, Revised 1985).

( ) For the protection of human subjects, the investigator(s) have adhered to policies of applicable Federal Law 32 CFR 219 and 45 CFR 46.

( ) In conducting research utilizing recombinant DNA technology, the investigator(s) adhered to current guidelines promulgated by the National Institutes of Health.

Diane Conti

Principal Investigator's Signature

12-11-98

Date

DAMD17-94-J-4500 FINAL REPORT
TABLE OF CONTENTS

1. INTRODUCTION

This document is Kensal Corporation's final report on a grant from the U.S. Army Medical Research and Materiel Command (DAMD17-94-J-4500) entitled "Dual-Use Telemedicine Support System for Pathology". Parallel work was done under a grant from the National Institutes of Health (2R44 GM444200-02A2) entitled "Image Handling System for Pathology and Telepathology" and a contract from Redstone Arsenal (DAAH01-95-C-R209) entitled "Dual-Use Intelligent Workstation for Medical Images."

Kensal has developed two telepathology workstations. The first was the Telemedicine Support System for Pathology (TSS), a Windows NT-based rapid prototype composed of many commercial-off-the-shelf (COTS) components. The commercial prototype is a G3/Power Macintosh-based PC Microscope (PCM) which is more compact than the TSS since the lensless scanner and traditional optics are contained in a computer tower. Both integrate the patented "lensless microscopy" with lensed full-color imaging (U.S. Patent 4,777,525).

During the development of the TSS, a library of pathology cases containing the scan of the entire coverslip and several corresponding high magnification images has been created. This research and development has spawned another tool which is called *Virtual Microscopy*. It is a self running multimedia CD-ROM used for viewing pathology cases generated by the TSS. For additional information, please see Kensal's 1997 annual report (Section 5, Virtual Microscopy) located in Appendix C.

2. RAPID PROTOTYPE TELEMEDICINE SUPPORT SYSTEM FOR PATHOLOGY (TSS)

The TSS consists of standard, commercial-off-the-shelf (COTS) components. A user interface permits production of a lensless "scout" image of the entire coverslip of a glass microscope slide (a virtual slide). Using the scout image as a reference, areas of interest (AOI) where finer detail is needed to complete the diagnosis can be magnified using traditional lensed microscopy. The TSS allows the user to transmit over ISDN (Integrated Services Digital Network) the scout image to a sister workstation for remote consultation. The remote consultant is able to select AOI from the transmitted scout image, the coordinates of those areas are sent back to the host TSS. There the high magnification images are captured and transmitted back to the remote consultant for diagnosis. The remote diagnosis and corresponding images are sent back to the host workstation for review and archiving for retrieval and/or transmittal at a later time. For additional information on the TSS not found in this section, see Appendices A, B, C, and D.

2.1 Imaging System

This task involved replacing the LSDA with a new three-channel RGB, 10,200-pixel, CCD imaging sensor with a fiber-optic faceplate that provides full slide coverage and to design a digital interface to the new Matrox image capture board.

2.1.1 The Design

The Kodak KLI-10203-A-BPA-A-O CCD with a MedOptics (Tucson, AZ) fiber-optic faceplate was mounted on a 7.235" X 2.7" printed circuit board which is raised and lowered onto the slide. All image processing and digital interface circuitry are present on this board. All digital timing for the KLI-10203 and interface timing are generated on one Altera PLD (EPM 7128-100) clocked by 30Mhz crystal oscillator. The digital drive signals needed by the CCD are generated by the EMP7128 and are sent to special drivers to provide wave shaping and voltage adjustments needed to drive the CCD during transfer phase and pixel read out.

Video processing starts with impedance matching buffers Q1, Q3, and Q4 for the three channels. The pixel data is then AC coupled and inverted by amplifier U3. The optical blacks from each pixel are clamped by D2. This ensures optical black uniformity from pixel to pixel across the full line of pixels. The second part of the clamping system is U2 providing a high impedance input from clamps. The clamped video now goes into a high-speed sample and hold amplifiers U10, U13, and U14 where the video portion is very precisely sampled and held until the next sampling, providing a high dynamic range video signal. The digital pulses that control the clamping circuit and sample and hold amplifiers are very precise and need adjusting. This is accomplished by digital delay lines DL1-DL5 that are adjusted by making solder bridges on the appropriate pads for the desired delay needed. After the sample and hold, the video is filtered and goes to the final amplifiers providing independent voltage adjustments on all three channels.

There is a simple digital interface to the Matrox image capture board. The CCD image board controls the timing and transmits a pixel clock that is used by the Matrox board to start and stop the analog to digital conversion. This signal is also controlled by an adjustable digital delay line for precise control and alignment of the analog to digital conversion process. There is also a start of line signal letting the Matrox board know when to start and stop a line.

2.1.2 Conclusions

The KLI-10203 board performed well. One major problem was with the fiber-optic faceplates. These faceplates are quite large and MedOptics had difficulty mounting it which resulted in nonuniformity in pixel to pixel amplitude and uneven illumination across the CCD. This problem can be solved by a different faceplate or by a different mounting method. The only other problem encountered was heat dissipation on the KLI-10203 board. This can be solved by breaking up the board and putting components with greater heat dissipation on another board. This will also generate a smaller board to be raised and lowered onto the slide.

2.2 Retrofit

The original version of the TSS employed a Kodak 13-micrometer line scan diode array (LSDA) and a Kodak evaluation electronics board. In late 1996, a dramatic improvement in low-resolution lensless microscopy was made possible by the introduction of the new EG&G Reticon RL4000P and the Kodak KLI-10203CA LSDA's that have diodes spaced on seven micrometer centers.

Due to problems experienced with Boeckeler Instruments, Inc. (Tucson, AZ), a subcontractor performing the retrofit, the TSS were retrieved and brfought in-house where functional status of the internal systems was assessed. It was determined that the systems required substantial work before being suitable for final delivery. Sizable problems included providing sufficient illumination for the KLI-10203 sensor, coding for the lensless scanner, and updating code for the support software (such as ISDN and voice recording). Systems to be completed before distribution included installation and modifications for the custom evaluation board, system wiring and hook-up, and system application coding.

2.2.1 Installing the Evaluation Board

The Kodak evaluation board was removed as well as the phase lock loop and other unessential electro-mechanical equipment. A custom power supply able to produce the necessary +6v, -6v, and 15v was installed in the existing power supply box. Custom cables were created to connect the power supply to the board, wire the pixel clock and line sync into the Matrox Genesis LC (LC) capture board and provide motion control for the DB9 serial cables. During installation it was noticed that the current illumination system did not illuminate the sensor fully. The previous Kodak sensor was 2 inches in length, the new Kodak sensor measures three inches in length.

Problems were compounded by the fact that the illuminations source in the old system was a tungsten filament lamp, which produces a predominately red light. The tungsten lamp was replaced with a white LED, and the assembly was moved closer to the sensor for illuminating more of the sensor length. The TSS is currently capable of capturing the entire coverslip, but not the entire sensor.

2.2.2 Installing KLI-10203

Kensal purchased sensors, KLI-10203, from Kodak and sent them to MedOptics for bonding of the faceplates. In the first faceplate design, the fiber optic material on the faceplate was surrounded by glass. This produced problems in the polishing and bonding procedures. The later faceplate design consisted of a solid piece of the fiber optic material. This faceplate design proved to be problematic as several faceplates were damaged by MedOptics. New methods are being tested to solve those problems.

2.2.3 Updating the TSS Application Code

Coding the TSS application to capture lensless scout images and high magnification images began at BII. The application had been designed to capture and display live video from the 3 CCD camera. However it was necessary to code the TSS application to capture, process, and store lensless images. Accessing the capture features of the Genesis LC card is done by using the Genesis LC API, called Mil-Lite. Mil-Lite uses a custom camera description file, created using the Matrox Intellicam program, to describe video timings, format, and size.

Lensless imaging on the TSS is not a perfect process and requires additional software correction including color registration, clear field correction and thumbnail generation. Color registration is necessary to align the RGB channels. As an artifact in lensless imaging with a faceplate, pixel to pixel illumination is nonuniform and requires software correction to remove visible streaking in the image. Taking a clear field of the image, the streaking is calculated out of the image. Additional code was written to generate the thumbnail image.

BII in error had reduced the screen size of the TSS application from specified resolution of 1024x768 to 800x600 in which displaying a 640x480 live image camera feed becomes a little cramped. Software modifications were dedicated to repairing such overlooked necessities. The GUI had to be altered to properly display overlay using the Mil-Lite. Marquees and overlays had to be written as they were simply overlooked by BII.

The sound recording and ISDN application portion of the TSS were updated to be compatible with the current version of the MFC.

2.2.4 Updating the TSS Hardware

Hardware was updated to support the TSS on Windows NT. A new mother board, additional memory, CD-ROM, jaz disk, CPU, sound cards, and case were installed. TSS code was updated for the new operating system.

2.3 DICOM

Kensal Corporation contracted with ImageLabs (Bedford, MA) and PlanetTools Solutions (Cave Creek, AZ) for production of software packages that could be integrated into the TSS and PCM system software for the purpose of converting the pathology slide data into a DICOM format. The strategy was to take advantage of completed and concurrent development on both the TSS and PCM. The first goal was to produce a software application for the TSS that could be efficiently

modified to support the PCM. Concurrent with this goal was for ImageLabs to support the parallel development by PlanetTools.

ImageLabs was unable to completely accomplish this task. Their final product to us was their *Shared Vision* viewer for radiology with another COTS software package added. Kensal asked for 24-bit resolution. ImageLabs was only able to produce 8-bit resolution. Their estimate for achieving 24-bit resolution was not available in our budget. Integrating this product into the TSS system software would have required many hours for producing a connecting code or the user would have to switch back and forth between the ImageLabs software and the TSS system software. This process leaves too much room for error.

Telecommunication and how it is achieved is the point of the TSS (Telepathology Support System) prototype. The method by which pathologists communicate between systems is crucial in the approval of the concept. Shared Vision provides "store and forward" the same way any application does, by using the operating system. In such a configuration the pathologist is responsible for supporting remote networking services and managing data files. The TSS and PCM attempt to shield the pathologist from the details of the operating system and file system by providing operations that perform telecommunication, file management, and archiving. If Shared Vision was implemented alongside the TSS application as planned, the only telecommunication benefit is real-time conferencing. Kensal made the decision to cancel any further development by ImageLabs.

PlanetTools Solutions was not only able to produce 24-bit resolution, but they produced it within the PCM system software where all interfaces are virtually invisible to the user. The PCM software is a much better product than the ImageLabs software package produced for the TSS and more than meets all of our requirements. All future workstations will use this software product. Plans are to convert the current G3 / Power-Macintosh version into a Windows NT version when funding becomes available for the next generation workstation. For additional information, please see Kensal's 1997 annual report (TPW Design Document) located in Appendix C.

3. COMMERCIAL PROTOTYPE PC MICROSCOPE (PCM)

The Windows NT-based, rapid prototype TSS produced for the U.S. Army is composed COTS components and occupies an entire desktop. The more compact G3 / Power Macintosh-based, commercial prototype PCM produced for the U.S. Army has been reduced in size. This extraordinarily simple and compact mechanism provides a PC (Personal Computer), the lensless microscope, and a lensed microscope in a single housing. Additional information not found in this section is located in Appendices C and E.

3.1 Imaging System

This task was to design and develop a video system consisting of a Texas Instruments (TI) TC217 CCD running at a maximum 14 Mhz pixel rate and an RL4000P single line CCD for slide scanning. The two CCDs are mounted on an optical assembly that will switch between two objectives, 10x and 40x. The focusing assembly moves the objectives relative to the TC217 CCD. The RL4000P's (LSDA) board is mounted to the same assembly and is raised and lowered by an electromechanical device.

A slide is inserted by the pathologist on a platform extended from the PCM. The microscope slide will be held firm, moved by an X,Y table assembly over a fiber-optic illuminator and under the selected objective lens to the LSDA for a complete slide scan. An illumination system was designed whereby red, green and blue LEDs will be turned on in sequence and delivered through a fiber-optic cable to a platform under each objective and the LSDA. This system will deliver enough light in red, green, and blue to produce high quality images through the

objectives and LSDA. The main PCM module contains all communications necessary to be controlled by a host computer.

3.1.1 The Design of the TC217 CCD Video Camera

Special serial drivers were designed to drive the three serial register gates of the TC217 at 14 Mhz. U1, U2, U5, and DZ1-DZ3 and a 74AC125 buffer provide level translation and high-speed high current drive for TC217 SRGs. Parallel transfer signals use the TI standard chip set U3 and U6, since parallel transfer speeds remain constant regardless of SRG speeds.

Video processing starts with impedance matching buffers Q5, Q7, and Q8 for the three channels. The pixel data is then AC coupled and inverted by amplifiers U8, U10 andU12. The optical blacks from each pixel are clamped by Q6. This assures optical black uniformity pixel to pixel across the full line of pixels.

The second part of the clamping system is U7, U9, and U11 providing high impedance input from clamps. The clamped video now goes into a high speed sample and hold amplifiers U15, U18, and U21 where the video portion is very precisely sampled and held until the next sampling, providing a high dynamic range video signal. The digital pulses that control the clamping circuit and sample and hold amplifiers are very precise and need adjusting. This is accomplished by digital delay lines DL1-DL5 which are adjusted by making solder bridges on the appropriate pads for the desired delay needed.

Next, the three video channels go into electronic switches U14, U16, and U22 which switch between the TC217 video and the RL4000P video. Next the video channels are filtered and sent to the three A to D converters U13, U17, and U20. Video now is in a digital format. Three channels for the TC217 and two channels for the RL4000P which are muxed by U30, U32, and U35 to one digital stream is now sent to the link controller PC board and on to the computer.

The digital timing is split between two Altera PLDs U25 clocked by an 80 Mhz crystal oscillator that supplies all timing for the TC217 and clamping. U34 controls all A/D timing, mux timing and link controller communcations.

The RL4000P circuit board mounted to the same optical assembly contains one RL4000P with a fiber-optic faceplate. All digital timing and communication with the TC217 board are in one Altera PLD U15 running at 5 Mhz. Digital transfer and pixel clocking are sent to level translating drivers U1, U2, U6 and then to the RL4000P. Two video output channels are coupled to impedance matching buffers Q1 and Q3 and then into inverting and output drive amplifiers U4 and U5. Due to the high output video levels from the RL4000P and the slow speed, no pixel clamping or sample and holds were used. The video is sent to the A/Ds on the TC217 board along with special pixel clocking for the A/Ds.

3.1.2 Conclusions

3.1.2.1 TC217 Camera

Although running the TC217 camera twice as fast as published specifications recommend, the camera worked adequately and produced usable pictures under controlled conditions inside the PCM. The TC217 was the only CCD at the time with 7-micron pixels. Now new CCDs have come on the market which would be better suited in this application and would improve picture quality. With the new CCDs available, two new avenues are available, one possibility is both Philips and TI have 1K x 1K frame transfer CCDs with 7-micron pixels that will run 30fps and would use less circuitry. Both of these new CCDs have single video output channels which eliminates any fixed pattern noise associated with multiple output channel CCDs. A new CCD just

coming on the market has 3.2-micron pixels in an array of 1.3 million and more. Since 4.3 microns is approaching cellular resolutions, direct imaging without the use of objectives may be possible. In any case, the camera would be redesigned in the next generation workstation.

The RL4000P with the exception of the fiber optic faceplate performs quite well. The fiberoptic faceplate was found to produce uneven illumination. This can be repaired by new bonding methods.

3.1.2.2 PCM Slide Delivery and Focus System

The slide insertion method currently employed in the PCM involves a platform that extends from the front of the unit. The slide is inserted and held by only one quarter of an inch at one end. The initial design concept presented many constraints because of size. There is no good way to insert a slide onto an extended platform without fingerprints and slide stability problems. This method will also solve other problems encountered. For example, when the focus motor is engaged, the whole floating optical assembly will shake and the center of view on the screen will move.

The focusing method used in the PCM is based on a floating platform containing the objective mounted on a quarter inch thick plate of aluminum which is supported by four spring clips. This is a lot of mass to move without any type of jitter showing up during live video. Two new focusing solutions are being considered for implementation should additional funding become available.

3.1.2.3 LED Fiber Optic Illumination System

The originally designed and fabricated LED illumination system was incapable of delivering enough illumination to the objectives -- primarily the 40x. The original design had nine one-candela LEDs arranged in red, green, and blue around an integrating sphere the bottom of which contains one end of a fiber-optic bundle. The other end ran to the custom aluminum platform under the objectives and the LSDA. The fiber-optic bundle, after entering the aluminum platform, was separated into three parts -- one .030 dia. bundle under the 40x; one .060 dia. bundle under the 10x; and, a .030" x 1.6" bundle under the LSDA. The first problem with this configuration was that the integrating sphere was at best 50% efficient, and the LED entry angle to the sphere was wrong. Adding to the problem, the light gathering end of the fiber-optic cable was .2" dia. and the hole in the top of the sphere was .32" dia. giving an estimated efficiency out of the sphere of only 15%. Looking at the distribution of light under the objectives we found that the 40x which needs the most light was getting the least in a .030 dia. spot. The 10x which needs less light then the 40x was getting almost twice the light from a .060 dia. spot, and the LSDA which needs the least light was getting the highest amount of light from the fiber-optic bundle.

It was obvious a change was called for so two parallel efforts were begun. One is directly connecting an optical fiber to an LED just above the die with optical epoxy, one each for red, green and blue. The fibers are then spliced together into one fiber. The length of this fiber produces a homogenizing effect on the light reducing the hot spots inherent to LEDs. Three of these assemblies are mounted in an aluminum platform under the objectives. This was to be a five-phase process. At the writing of this final report only phase two had been completed. The second effort is the brute force method. Taking three large LED arrays (TO66 packages) -- one red (400mw output), one blue (40mw output), and one green (200mw output) -- mounting them on a new integrating sphere and using the same fiber-optic assembly. At the writing of this report, delivery of the new sphere was delayed passed the deadline. It is believed that both of these methods will have a high degree of success.

3.2 Control Electronics

This section describes the electronic hardware contained within the PCM from a general, top level perspective. Please consult the respective design documents located in Appendix E for a more detailed description of any of the individual components.

3.2.1 PCM Electronic Hardware

Referring to Figure 1 below, the PCM System is composed of two main blocks: The PCM Optics Assembly and the Macintosh 9500 Computer. Within each block are several custom designed printed circuit boards (PCBs) that give the PCM its functionality.

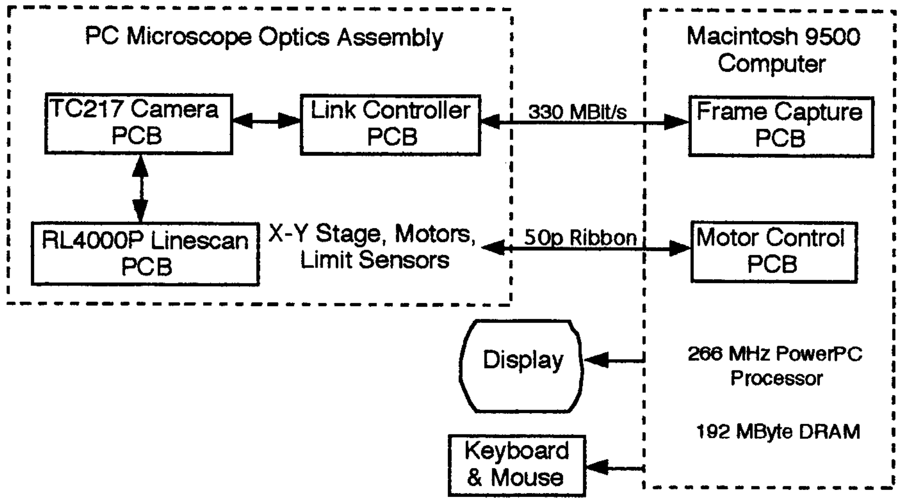


Figure 1 - System Hardware Block Diagram

3.2.1.1 PCM Optics Assembly

The PCM Optics Assembly is a fully custom assembly for microscope slide scanning and high magnification inspection. Within this unit are three custom PCBs: (1) The RL4000P Linescan PCB contains the EG&G RL4000P Linescan Sensor. The RL4000P provides low mag guide images of the microscope slide, and performs analog signal conditioning and timing generation necessary to digitize RL4000P video. The actual digitization is handled on the TC217 PCB. (2)The TC217 Camera PCB contains the TI TC217 Array Sensor. The TC217 sensor provides high magnification images of the microscope slide. This PCB performs analog signal conditioning, analog-to-digital conversion (ADC), and timing generation necessary to digitize TC217 and RL4000P video. Digitized data is sent to the Link Controller PCB for transmission to the Macintosh host. (3) The Link Controller PCB provides communication over a high speed Fibre Channel serial link to the Macintosh host. The PCB contains lookup tables (LUTs) for image correction. It also provides drive electronics for the red, green and blue LEDs used for slide illumination.

3.2.1.2 Macintosh 9500 Computer

From the outside, the Macintosh 9500 Computer assembly appears to be an OEM Apple Macintosh 9500. It has been upgraded for this application to include a 266 MHz PowerPC processor (the OEM processor is 120 MHz), and 192 MBytes of DRAM (OEM, 16 MBytes). Also, two custom PCI cards have been designed to interface with the PCM Optics Assembly: (1)The Frame Capture PCB has a high speed Fibre Channel serial link, 12 MBytes of Graphics DRAM, and special circuitry for viewing images with over a million pixels at frame rates up to 30 frames per second. (2) The Motor Control PCB contains controllers and pulse width modulated

(PWM) drivers for 4 axis of motion control, along with sensor signal conditioning and circuitry for driving solenoid actuators.

External to the Macintosh 9500 computer are a high resolution 17" monitor, and a standard keyboard and mouse.

3.2.2 High Level Design - Frame Capture PCB & Camera Head Electronics

The CHE (camera head electronics) is composed of three PCB's: RL4000P, TC217 and Link Controller PCB's. Since it was written during the design process, the language is from the perspective of a design in progress. Rather than summarize this document and take the chance of introducing errors or ommisions, it has been included it in its entirety. It has been fully updated to reflect any design changes that were made subsequent to the original release of this document. Thus, the information contained within this document matches the actual electronics developed for this project.

Detailed information from a programmer's standpoint on the functional operation of the Frame Capture PCB is provided by the document entitled "Frame Capture PCB Register Definitions, Rev D. Similar information for the Link Controller PCB and Camera Head Electronics is provided by the document "Camera Head Register Definitions, Rev D. Schematics and PLD listings for this PCBs discussed in this document are also included in Appendix E. More information about the Fibre Channel interface referenced herein may be obtained in the following data sheet: "CY7B923/CY7B933 Hotlink™ Transmitter/Receiver" available from Cypress Semiconductor, San Jose, (408) 943-2600.

The goal of this design effort is to develop and produce a "Fast Frame Rate Camera" for a medical pathology workstation (PCM). The camera uses a TI TC217 CCD sensor coupled to a Macintosh 9500 computer via a custom PCI interface board. "Fast" is defined as "no perceptible delay" when it comes to single frame snap shots, and approximately 30 fps (frames per second, monochrome) / 10 fps (RGB) for a 1134h x 972v image. A second sensor, an RL4000P linescan array is also coupled into the camera, but only one sensor can be active at a time. The RL4000P contains 4096h x 1v pixels. The microscope slide is continuously moved under the sensor such that 8000 lines x 3 colors (RGB) are exposed and read in 10 seconds or less.

The purpose of the report is to explain the design chosen to implement the intended functions. Other designs may be possible for similar functions, but, in most cases, the reasons why a particular design was chosen over another are beyond the scope of this document.

3.2.2.1 General Topology

3.2.2.1.1 Overall Structure

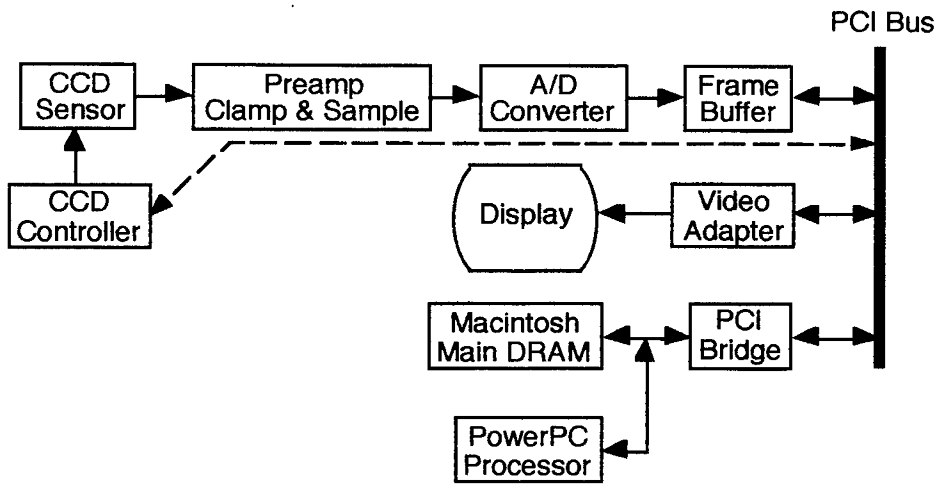


Figure 2 - General Topology

A block diagram of the system containing the Fast Frame Rate Camera is depicted in Figure 2 above. A summary of each of the blocks follows: (1) The CCD Controller generates all CCD, clamp and sample, and A/D converter timing signals. Since video modes and exposure rates are indirectly set by this block, a link must exist to the PCI bus for control/status information (indicated by the dotted line). (2) The CCD Sensor includes TC217 CCD and related driver IC's. (3) The Preamp / Clamp & Sample contain the analog signal conditioning required to prepare the CCD's output for digitization. (4) The A/D Converter is a suitable analog to digital converter. It needs to be at least 8-bits wide. Since the TC217 provides 3 output channels, there will be 3 A/D converters, each running at 1/3 of the overall pixel rate. We are using 10-bit A/D's for this function. (5) The Frame Buffer is a holding area to store video image information. (6) The Video Adapter is an Apple PCI Adapter with 4 MB of VRAM (supplied on an OEM basis from ATI). A complete image or a part of an image within the Frame Buffer can be transferred directly into Video Adapter VRAM without crossing any PCI bridges. (7) The PCI Bridge is a custom chip developed by Apple called a "Bandit" which resides on the Macintosh 9500 main logic board. (8) The Macintosh Main DRAM is the main program/data store used by the Macintosh processor. Images within the Frame Buffer can alternately be transferred to main DRAM. This would most likely be the case when the line scan array is being used. (9) The PowerPC Processor is a PowerPC 604 G3 RISC processor running at 266 MHz.

The CCD Controller and Preamp must reside in close proximity to the CCD Sensors (TC217 and RL4000P), which, in turn reside physically close to optical image planes. These units are collectively considered to be the "Camera Head" and may reside some distance (from 1-3 meters) away from the PCI bus in the Macintosh. Thus, the first design issue to tackle is how to get the video information from the camera head to the Macintosh chassis. One approach leaves the video in analog form and sends it via three 75 Ohm transmission lines to the PCI interface card, where the A/D Converter and Frame Buffer would be located. An alternative option places the A/D converter in the camera head and instead sends digital data over a copper or fiber optic cable to the Frame Buffer on the PCI card.

3.2.2.1.2 Analog Connection

The analog method was considered, then discarded in terms of a digital topology outlined below. The major "cons" were:

- It would be very difficult to send wideband high resolution video (settle to 0.1% in 70 ns) over a cable.
- Timing becomes very critical at both ends. A fair amount of circuitry would be required to receive and properly digitize the video. It would be difficult to dynamically reconfigure the frame capture board for a different sensor (e.g.. RL4000P).
- Ground loops are inevitable with this arrangement. The cable acts as an antenna, so the camera head would be susceptible to ESD and may generate RFI. Due to the wideband nature of the analog signal, minimal filtering could be used to reduce noise.

3.2.2.1.3 Digital Connection - Fibre Channel

An alternative to analog video transmission is to implement an all digital link. The video is digitized, encoded (8B/10B), and sent over a copper or fiber optic channel to the PCI card. The major "pros" are:

- A 10-bit A/D converter can be used. The extra two bits can be used to provide additional dynamic range to allow offset, gain, and gamma correction to be performed in the camera head before 8-bit video is sent to the PCI card.
- For copper media, wideband pulse transformers would provide high common mode noise immunity. A fully shielded connector and cable assembly would minimize ESD susceptibility / RFI generation. The fiber optic media provides the same benefits described above, only heightened due to the optical nature of the link.
- The emerging Fibre Channel specification is causing many IC companies to produce transceiver solutions that can be readily adapted to our use.
- Although not a factor in our setup, a fiber optic link would allow much longer distances between camera head and computer (on the order of 1 km).

3.2.2.1.4 Data / Command bytes

The emerging fibre channel specification allows for the transmission of ~11 user specified codes along with the 256 possible values a data byte may contain. We can use this command channel to communicate mode or status information. It is anticipated that the link will be bidirectional, so mode and gamma table information can be sent to , and video data received from the camera head.

3.2.2.1.5 Fiber Optics

Low cost fiber optic transceivers are available that use LEDs to send data up to 320 MBits/s. Typically, anything above 320 MBits has required the use of expensive laser diodes instead of LED's. Optical Communication Products (OCP) is developing a 566 MBit link that will be LED based.

3.2.2.1.6 Coax w/Transformers

Part of the fibre channel specification allows for transmission over 150 Ohm shielded twisted pair (STP) or 75 Ohm coax in order to reduce the cost for limited distance links. Wide bandwidth pulse transformers can be used to provide Galvanic isolation and high common mode noise rejection. Transformers are most commonly used at data rates of 133 or 266 MBits/s, although literature suggests that 531 MBit links are possible using 75 Ohm coax. An evaluation board from Cypress Semiconductor uses transformers for 330 MBits/s 150 Ohm STP. Our link will operate at 330 MBits/s and will use the same configuration the Cypress evaluation board uses. We will use a specially balanced twisted pair cable from W.L. Gore specifically designed for wide bandwidth (> 1 GBaud) fibre channel applications.

3.2.2.1.7 Error Detection & Correction

In network applications, the reliability of data transmissions must be guaranteed. To this end, chip sets are available to encode data streams using the CRC-32 algorithm. Our application is atypical on several counts. First, our cable length will be between 1-3 m, resulting in a significantly higher signal to noise ratio than found in most network environments. Consequently, our bit error rate (BER) will be exceedingly low (estimated < 10^{-12}). Second, we *will* be able to detect many 1-2 bit errors due to the nature of the 8B10B coding standard in which only ~267 of a possible 1024 codes are used. During link development, we will use this error detection ability to ensure that our error rate is sufficiently low. Thirdly, in video applications such as ours, one or even two bit errors are non-issues as long as they don't occur very often. For the above reasons, we will not be incorporating CRC encoding/decoding.

3.2.2.2 Camera Head Electronics (CHE)

3.2.2.2.1 Structure

Figure 3 below depicts the structure of the CHE.

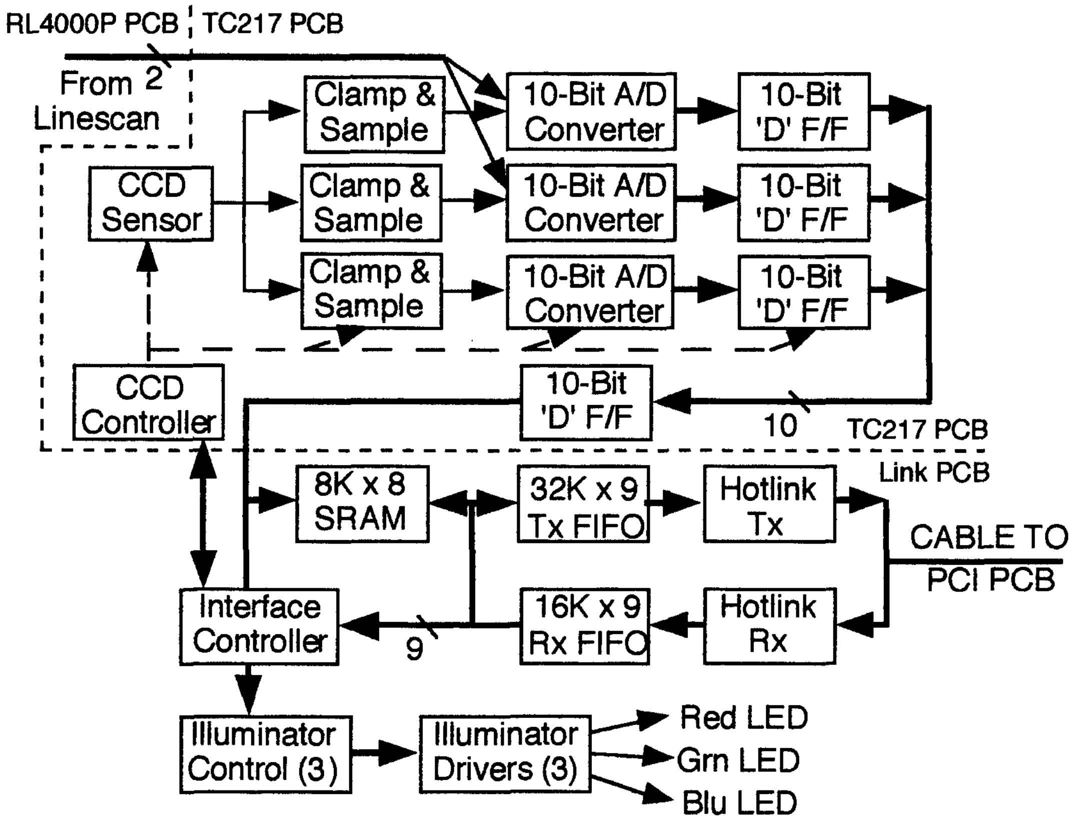


Figure 3 - Camera Head Block Diagram

The CHE are contained on three PCBs: RL4000P, TC217, and Link Controller. The RL4000P PCB holds the RL4000P sensor, biasing supplies, and analog signal conditioning circuitry. It is not implicitly shown on the block diagram above, but connects to it from the upper left corner. The TC217 PCB contains the TC217 area sensor and biasing supplies, analog signal conditioning, RL4000P video mux, triple A/D converters and high speed 'D' flip-flop registers. The Link Controller PCB (bottom half of the block diagram) holds the fibre channel interface, SRAM LUT's, and LED illuminator controls. The following description details the operation of the CHE without taking into account the artificial demarkation of the three PCB's that comprise it.

The EG&G RL4000P linescan feeds video data into two of the three video channels used by the TC217 via analog switches between the Clamp & Sample and A/D Converter blocks (not shown in the block diagram). The 10-bit A/D converters run at 13.3 MPixel/s (for each of three channels of the TC217) or 5 MPixel/s (for each of two channels of the EG&G RL4000P linescan array). The 10-bit latches time-multiplex the three pixel streams, forming a single stream of up to 40 MPixel/s burst. This stream feeds an 8K x 8-bit SRAM which is configured beforehand to perform the desired gain/offset/gamma correction for either the CCD or linescan array. A total of 5K of the RAM is actually used (3K for the TC217 and 2K for the RL4000P). This correction process results in a 40 MByte/s data stream (where one pixel is now one byte wide), which is written into the transmit FIFO. Data is clocked out of the transmit FIFO at 33 MByte/s and sent to the PCI board via the Cypress Hotlink encoder / transmitter. A 32K word deep transmit FIFO was chosen to insure that overflow doesn't occur. Writes to the FIFO occur only during the active pixel horizontal readout, whereas FIFO reads can occur during active or dummy/dark pixel readout, horizontal shift, image area to storage area transfer, and dark lines.

Before images are transferred, the camera head SRAM must be loaded by the PCI Frame Capture PCB with the appropriate gain / offset / gamma correction tables. Control and status information may also be exchanged using the bidirectional link.

The capability to perform Built In Self Tests (BIST) is incorporated into the Hotlink circuitry. This allows a specified worst case pattern to be circulated from the transmitter to the receiver. A 6-bit error count register is maintained by the Link Controller to totalize errors. This register may be queried by the PCI Frame Capture PCB after the test concludes. BIST mode is entered/exited via a jumper on the Link Controller PCB. Two green LEDs on the PCB indicate receive or transmit activity. A red LED latches 'on' when an error is detected by the Link Controller. Errors may occur from several sources. The Frame Capture PCB can interrogate a register within the Link Controller PCB to ascertain the exact cause of the error.

When the PCI PCB is not actively receiving video, the CCD Controller is in a mode where the TC217 sensor (or RL4000P, if selected) is continually being read out and digitized (but data is not written into the Tx FIFO). This is a crucial design decision that greatly simplifies the camera head design. Since the sensors are continually being read out, the concerns about fast clearing, exposure control and timing, and single shot modes and control are gone.

It should be noted at this juncture that the PCI Frame Capture PCB (described below) is relatively generic in nature. The PCB's comprising the CHE are not. Should one desire to use the PCI Frame Capture PCB with another sensor or linescan array (e.g. Kodak), another set of camera head electronics would have to be designed and fabricated. Once the data is digitized, corrected, and placed into a 33 MByte/s stream, the Frame Capture PCB can grab and display it easily. Effectively, the PCI Frame Capture PCB neither knows nor cares where the pixel stream came from.

3.2.2.2.2 Timing

If only the active video of the TI TC217 array is transmitted to the Frame Capture PCB, our goal is to transfer approximately sixty 1134h x 488v fields every second. Assuming no interframe gaps in the transmission, this would require a 31.7 MByte/s link. The link we will use is capable of a sustained 33 MByte/s rate, so we'd expect the transmission to be nearly continuous.

The bandwidth requirements of the RL4000P are substantially less demanding. Our goal is to read 8000 lines x 3 colors (RGB) in 10 seconds or less. This computes to an average rate of 9.4 MBytes/s.

3.2.2.2.3 PCI Frame Capture PCB

The PCI Frame Capture PCB is responsible for storing and displaying a quasi-continuous 33 MHz stream of pixel data. The block diagram for this complex function is depicted below.

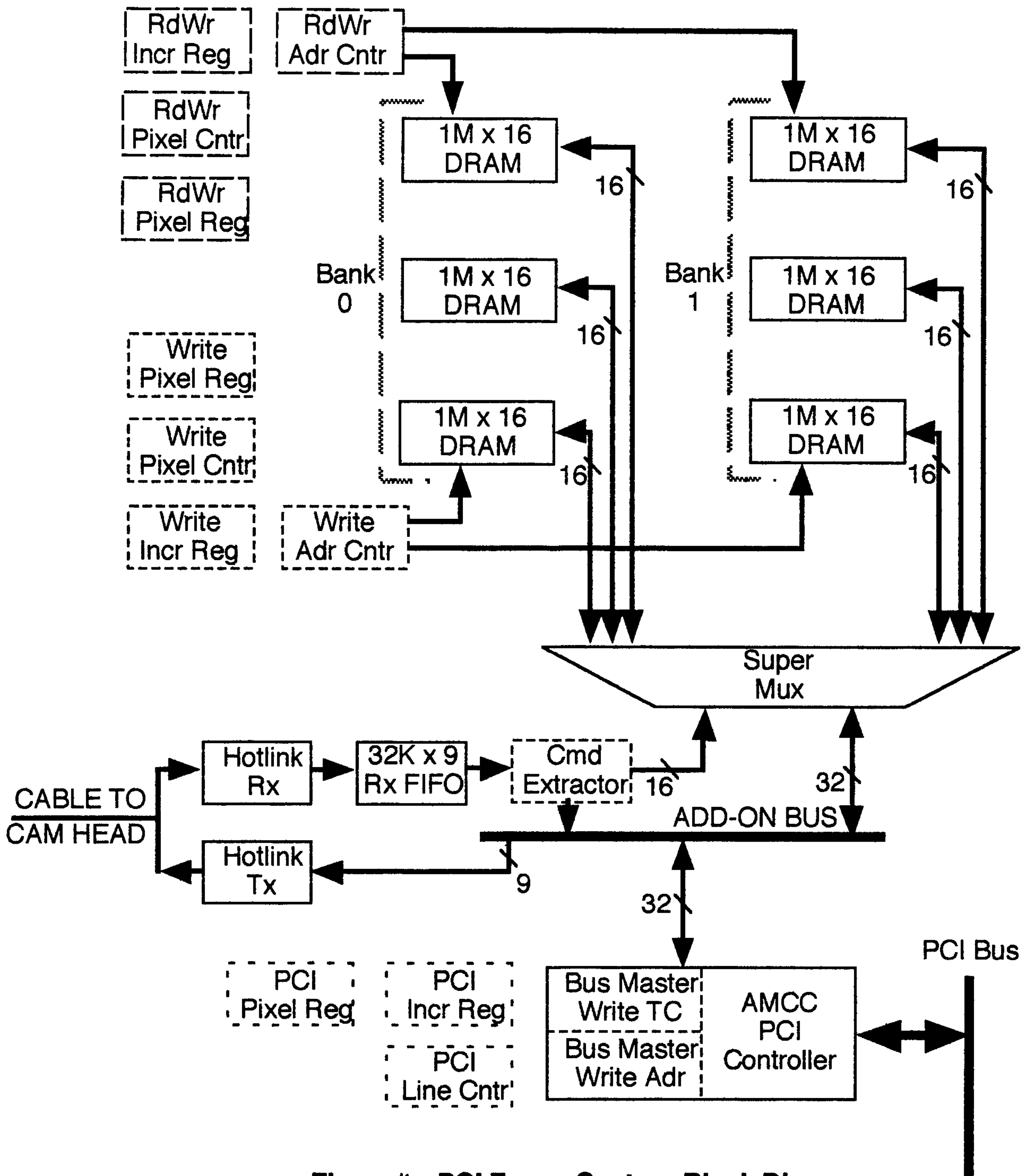


Figure 4 - PCI Frame Capture Block Diagram

3.2.2.2.3.1 Structure

Referring to Figure 4 above, note that there are two banks of DRAM, bank 0 and bank 1. Each bank contains a total of 6 MBytes of RAM, for a total of 12 MBytes on-board. Each 2 MB DRAM is capable of holding two complete interlaced fields (odd and even) from the TC217 with room to spare (unfortunately, the 1134 x 486 x 2 format of the TC217 is 5% larger than the space available in a 1MB memory).

The three DRAMs in each bank hold information for the three colors red, green and blue. Thus, at the end of 6 exposures (odd/even fields of R, G, B, at 60 fields/s), a complete RGB image is contained in a DRAM bank.

Shortly after data from one frame is stored, data from the next appears in the receive FIFO. After a brief reconfiguration from the PowerPC processor, data can be read out and stored in the opposite bank. While this is occurring, the assembled image can be transferred from the first bank across the PCI bus either directly to the video graphic board or the CPU's main memory for further processing. Please consult the "Interrupts & Timing" section below for timing information specific to the TC217 and RL4000P sensors.

3.2.2.3.2 Hotlink Controller

The Hotlink Controller (not shown in the block diagram) manages the interface between the Rx FIFO and the Tx and Rx Hotlink EnDecs (Encoder/Decoders). The controller will run at the maximum design frequency of the endecs, which is 33 MHz. The controller will be placed within the same Altera Device used to contain the Write Controller and Command Extractor.

3.2.2.3.3 Command Extractor

First, a little background before jumping into the Command Extractor function. The 330 MBits/s serial bit stream is received by the Hotlink receiver, where it is decoded to produce a 33 MWord/s data stream. Each word consists of 8 data bits and a command identifier bit. If the command bit is '0', then the 8-bits are interpreted as data. If the command bit is '1', then the 8 data bits refer to one of the 10 or so commands that can be sent across the link. Note that only about 10 of the possible 256 bit combinations are valid.

Data will be transferred across the interface in packets. Each packet of data will be preceded and followed by specific command bytes used to identify the data within the packet. These are outlined in the Data Traffic section below.

Command / data words are synchronously written into a 32K x 9-bit FIFO by the Hotlink Controller (see above). The Command Extractor block reads data from the FIFO at 36 MWords/sec and performs 3 main functions: 1) forms 16-bit data words from two 8-bit data bytes. These 16-bit words are written using Fast Page Mode writes into the DRAM of the selected bank and color, 2) stop writing DRAM and generate a PCI interrupt when a command (ie. non-data) word is received, and 3) allow the PCI bus to access the command/data stream on a word by word basis.

3.2.2.3.4 Write Controller

The write controller consists of the Write Pixel Register, Write Pixel Counter, Write Increment Register and Write Address Counter blocks. It also works closely with the Command Extractor block described previously. Before data is to be written to a bank, the Write Pixel, Write Increment and Write Address registers must be loaded from the PCI bus. The Write Pixel register contains the number of DRAM writes per video line. The Write Increment register contains a number to be added to the Write Address Counter at the end of a line in order to point to the start of the next line. The Write Address Counter initially contains the address of the first pixel of a given field.

When the Write Controller is started, the Write Pixel Counter is initialized with the value contained in the Write Pixel Register. Data from the Command Extractor is written to the active bank at the address specified by the Write Address Counter. The Write Address Counter is incremented and the Write Pixel Counter is decremented. When the pixel counter reaches zero, the end of a line is indicated. The contents of the Write Increment register are added to the Write Address counter to form the first address of the next line. In this way, interlaced video can be stored in memory properly.

These blocks will be placed in a single Altera PQFP programmable device clocked at 36 MHz. It is important that this controller processes Rx FIFO data faster than the Hotlink Controller

can write to the FIFO. There will be periods between fields when the PowerPC processor is setting up the next transfer that no data will be read from the FIFO even though data is pouring in. When the Write Controller is subsequently started, it must run faster in order to catch up and empty the FIFO.

3.2.2.3.5 Read/Write Controller

The Read/Write Controller consists of the Read/Write Pixel Register, Read/Write Pixel Counter, Read/Write Increment Register, and the Read/Write Address Counter blocks. It functions in a complementary fashion to the Write Controller except that data is either read from (typically) or written to (during diagnostics only) the DRAM from the PCI bus instead of the Hotlink Rx FIFO. Note also that the Write Controller and Read/Write Controller cannot access the same bank at the same time. There just isn't enough bandwidth to allow pure dual port access.

3.2.2.3.6 Super MUX

The "Super MUX" was so named because of the complex multiplexing that must occur in order to steer video data around from/to the proper locations. During frame capture, data is typically received for one color at a time (the Kodak RGB linescan array is a notable exception). However, when video data is written to a video graphics board, entire RGB pixels must be processed at one time. Thus each of the three DRAMs in a bank will be accessed at once during readout.

Two pixel sizes will be supported by the MUX. In 32-bit-per-pixel mode, a pixel is composed of three 8-bit RGB values and an 8-bit placeholder (filled with 0x00). If 16-bit-per-pixel mode is used, a pixel is composed of the 5 most significant bits of red, green and blue, along with an "alpha" bit of '0'. Two 16-bit pixels are written with each 32-bit PCI cycle. Note that the 8-bit-per-pixel indexed video mode is not supported by this board.

Each of the pixel sizes can operate in either RGB or monochrome mode. In RGB mode, data is taken from the three DRAMs representing RGB and an RGB pixel is fashioned. In monochrome mode, data is taken from the R, G, or B DRAM and duplicated to form an RGB pixel. The result is an image with either 32 or 256 shades of gray (depending on whether the board is using 16-bits/pixel or 32-bits/pixel).

The Super MUX operates at the PCI bus clock rate of 33 MHz.

3.2.2.3.7 PCI Controller

The PCI Controller function refers to the circuitry required to interface the PCI bus with the rest of the circuitry on the frame capture board. Part of the function is contained within an Altera CPLD, and part is contained within an AMCC S5933Q PCI interface IC. The function contains the following blocks: PCI Pixel Register, PCI Increment Register, PCI Line Counter, Bus Master Write Transfer Counter and Bus Master Write Address Counter. It functions similarly to the Write Controller, except that the destination is PCI memory, as opposed to onboard DRAM. Note that the Bus Master Write Transfer Counter and Bus Master Write Address Counter are contained within the AMCC 5933Q PCI interface IC but are an integral part of the overall PCI Controller function.

Since there is no FIFO buffering between the DRAM banks and the FIFO within the AMCC PCI interface IC, the PCI Controller must operate closely with the Read/Write Controller and Super MUX when access to/from the DRAM is desired. For example, when the FIFO within the AMCC chip reports full status, the pipeline must immediately be frozen *on the same clock cycle* or else data will be lost. This represents somewhat of a design challenge considering all of the modes that the Super MUX can operate in, but it is possible to accomplish.

3.2.2.4 *Data Traffic*

3.2.2.4.1 *Commands*

As stated above, data from the Hotlink transmitter and receiver IC's consists of an 8-bit data byte and a 1-bit command identifier. When the command identifier bit is active, only about 10 of the 256 possible 8-bit codes are valid – the rest are invalid and unusable. Of the 10 valid codes, the frame capture board uses 4, outlined in Table 1 below.

Name	Direction	Description
WriteHdr	From PCI	Packet header to request write to Camera Head
ReadHdr	From PCI	Packet header to request read from Camera Head
DataHdr	From Cam	Packet header used for transmission of data from Camera Head.
EndPkt	Both	Universal packet trailer.

Table 1 - Commands in Data Stream Between Camera Head and Capture Board

When the frame capture board wants to write data to the camera head, it sends a WriteHdr command, followed by a Data Selector byte (described below), followed by the data to be written, and, lastly, terminated by an EndPkt command. The camera head does not send a respond to the packet sent.

Similarly, when the frame capture board wants to read data from the camera head, it sends a ReadHdr command, followed by a Data Selector byte, and terminated by an EndPkt command. The camera head then replies by sending a DataHdr command, followed by a Data Selector byte, followed by the requested data, and, lastly, terminated by an EndPkt command.

3.2.2.4.2 *Data Selector Byte*

The Data Selector Byte identifies the data to be written or read. The various data types are enumerated in Table 2 below.

Name	Description
CtrlStat	Control/Status Register
ErrCnt	Error Counter Register
Delay	Delay Registers
LED	LED Intensity Registers
LUT	Gain/Offset/Gamma LUT's
Even	Even TC217 Field
Odd	Odd TC217 Field
Line	RL4000P Linescan Frame

Table 2 - Data Selector Byte

The Control/Status and ErrCnt registers are single byte entities. Their bit definitions, along with those for the Delay and LED Intensity registers, are contained in the document "Camera Head Register Definitions". The LUT is an 8K by 8-bit block of data organized in 1K blocks. Whenever

LUT data is written or read, the entire 8K x 8 LUT is transmitted. The organization of the LUT is depicted in Table 3.

Address Range	Description
0x0000 - 0x03FF	TC217 Channel 1 LUT
0x0400 - 0x07FF	TC217 Channel 2 LUT
0x0800 - 0x0BFF	TC217 Channel 3 LUT
0x0C00 - 0x0FFF	Reserved (unused, should be 0x00)
0x1000 - 0x13FF	RL4000P Channel 1 LUT
0x1400 - 0x17FF	RL4000P Channel 2 LUT
0x1800 - 0x1BFF	Reserved (unused, should be 0x00)
0x1C00 - 0x1FFF	Reserved (unused, should be 0x00)

Table 3 - LUT Organization

The TC217 Fields (Even and Odd) are each 551,124 bytes in length and represent 486 lines of 1134 pixels per line. Note that dark lines, dark pixels, and dummy pixels are normally not transmitted in the data stream. It is possible to request data frames containing these pixels, however, for diagnostic purposes. The RL4000P Frame (Line) data constitutes a block of 4096 bytes. Similarly to the TC217, dark and "isolation" pixels are not transmitted to the frame capture board (except upon explicit request).

3.2.2.4.3 Interrupts & Timing

As a general rule, the generation of interrupts should be kept to an absolute minimum, since there is a significant overhead in servicing them. PowerPC processor intervention is required under two general conditions: 1) Some setup of the Frame Capture PCB is required before data transfer can proceed, and, 2) A real-time event has occurred in the Camera Head requiring PowerPC processor intervention.

The only real-time event of significance is the completion of a transfer from the image area to the storage area of pixel data in either the TC217 or RL4000P. At this point it is necessary to change exposure color or move the microscope slide to another location, or both. Fortunately, the transfer completion event occurs immediately before readout commences, so the transmission of the DataHdr and Data Selector Byte (Odd, Even, or Line) can be made to coincide with that event. So, with one interrupt, the PowerPC can change position and/or illumination for the next field, as well as setup the Frame Capture PCB to receive the previous exposure's data. Once data transfer is started, the next interrupt will occur roughly 1/60th second (16.6ms) later when the EndPkt command is received at the end of the video data packet.

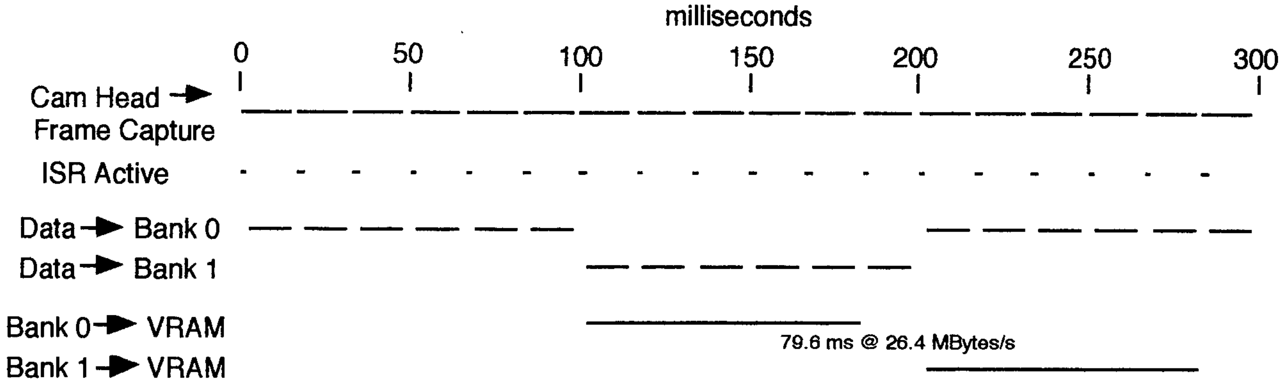


Figure 5 - TC217 Frame Timing

The timing diagram in Figure 5 above illustrates the double buffered nature of transfers from the TC217 in the Camera Head through the PCI Frame Capture PCB to the video RAM on the ATI Graphics Board. The top line indicates that the data stream from the Camera Head to the Frame Capture PCB is nearly a continuous 33 MBytes/s. The six streams of data correspond to odd/even fields of R, G, & B. The second line depicts when the 9500's Interrupt Service Routine (ISR) is activated in order to setup an incoming transfer. During the ISR, data is temporarily stored in a FIFO on the Frame Capture PCB. The "Data -> Bank X" dashes can be shorter than the input stream dashes because the FIFO read rate is 36 MBytes/s vs. the write rate of 33 MBytes/s from the Hotlink receiver link. The long dash in the "Bank X -> VRAM" lines indicate the amount of time needed to transfer one frame of data assuming a PCI transfer rate of 26.4 MBytes/s (see the Performance Estimation section below for why 26.4 MBytes/s was used). 26.4 MBytes/s is a conservative lower limit. The actual rate will be higher. All of the above data assume a final pixel depth of 16-bits.

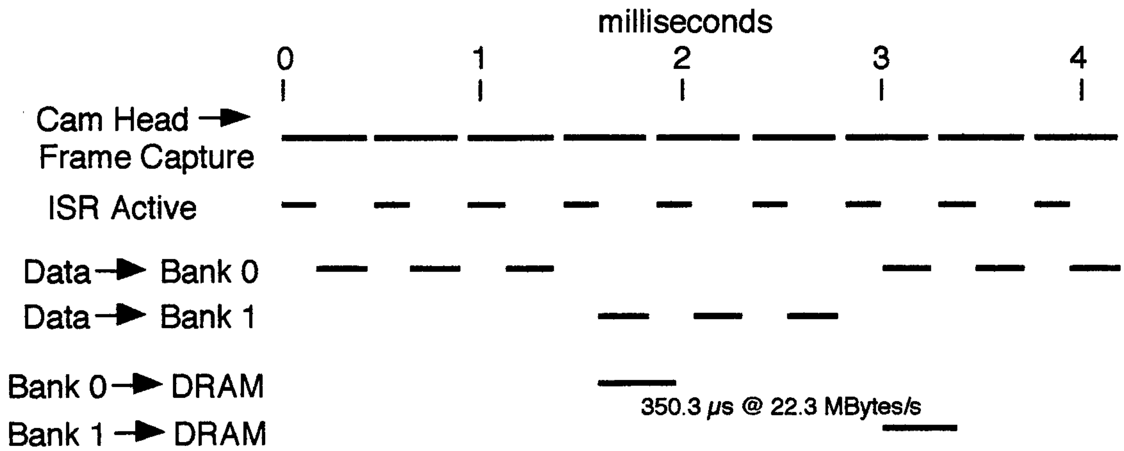


Figure 6 - RL4000P Frame Timing

The timing diagram in Figure 6 above illustrates the timing for a RL4000P linear array. Note that complete *lines* of the RL4000P are double buffered, as opposed to complete *frames* on the TC217. Also, the burst data rate from the Camera Head is about 10 MBytes/s vs. 33 MBytes/s with the TC217. Like the TC217, the RL4000P requires 3 exposures to collect R, G and B data for each line. Finally, note that the final destination for the data is the Macintosh main DRAM. vs. the VRAM on the ATI Graphics Board. Transfers to DRAM use the Apple Bandit PCI bridge IC, which limits PCI throughput to 22.3 MBytes/s. All of the above data assume a final pixel depth of 16-bits.

3.2.2.4.4 Windowing

Windowing refers to the ability to write only a subset of the frame's data into the PCI memory space. This feature is crucial if it is required that the Frame Capture board write directly to a PCI graphics board. The Frame Capture PCB will allow rectangular windowing. Thus, any rectangle within the image can be extracted and placed anywhere on the Macintosh's monitor screen. There will be some limitations, however. For example, in 16-bit/pixel mode, the rectangle must begin and end on even pixel boundaries (assuming pixels are numbered starting at zero). In 32-bit/pixel mode, there are no such limitations.

3.2.2.4.5 Macintosh 8500 Platform

One might consider using the Macintosh 8500 platform for the fast frame rate camera. It has 3 PCI slots, built-in video support, and is less expensive than the 9500. However, there are several negatives associated with the platform for this application. First and foremost, access to the built-in video controller from the PCI bus must occur through the "Bandit" PCI bridge ASIC. This IC currently only allows two 32-bit transfers per cycle maximum when not filling the processor's cache. This limits PCI bandwidth to approximately 22.3 MBytes/s, from a theoretical maximum of 132 MBytes/s. Other factors are 1) only 3 vs. 6 PCI slots in 9500, 2) 256 KB vs. 512 KB level 2 cache in 9500, 3) video support in 8500 appears to be non-accelerated vs. the accelerated support obtained by the Apple OEM ATI board.

3.2.2.4.6 Estimated Performance

From previous tests involving the PCI Morphology PCB, it was demonstrated that bus master mode transfers could occur from the Morphology PCB to the PowerPC's main DRAM memory at 22.3 MBytes/s. The limiting factor was the "Bandit" PCI bridge IC on the 9500's main logic board. A test was conducted using the same Morphology board to write instead to the Apple OEM PCI Graphics board. The Morphology board was plugged into the same PCI bus segment that contained the Graphics board. Master mode transfers were measured at 26.4 MBytes/s. This time, the limiting factor was the Morphology PCB. The state machine design requires 5 PCI cycles per 32-bit write to the S5933Q's FIFO. Unfortunately, at this time it is impossible to gauge the maximum throughput the PCI Bus / Graphics Adapter board can handle. All we can surmise is a lower limit of 26.4 MBytes/s. We require approximately 22 MBytes a second throughput in order to refresh a full image at 16-bits/pixel. The throughput requirement doubles to 44 MBytes/second for 32-bit pixels. The Frame Capture board will be designed to transfer a 32-bit word every PCI cycle in 32-bit/pixel mode, so the throughput bottleneck should not rest with the Frame Capture board.

3.2.2.4.7 Attained Performance

Transfers between a Frame Capture Board and a Matrox graphics adapter on the same PCI segment have been measured at 48.3 MBytes/second for 16-bit pixels, and 17.5 MBytes/second for 32-bit pixels. At this point, the discrepancy between the two transfer rates cannot be explained. Perhaps the video board is optimized for 16-bit pixel depth and thus represents the bottleneck. Since we are operating exclusivly at 16-bit depth, it was determined that this mystery could remain unsolved at the present time.

Transfers from a Frame Capture Board to main memory across the PCI bridge have been measured at 34.1 MBytes/second for 16-bit pixels and 31.6 MBytes/second for 32-bit pixels. Thus, the performance demonstrated by the completed Frame Capture PCB has exceeded expectations and is thoroughly adequate for real time video capture and display of high resolution images.

3.2.3 High Level Design - Stepper Motor PCB

This document was written during the design process. The language is from the perspective of a design in progress. Rather than summarize this document and take the chance of introducing errors or ommisions, it is included it in its entirety. It has been fully updated to reflect

any design changes that were made subsequent to the original release of this document. Thus, the information contained within this document matches the actual electronics developed for this project.

Detailed information from a programmer's standpoint on the functional operation of the PCB is provided by the document entitled "Motor Control Board Register Definitions, Rev B". Since a majority of the board's functionality is provided by a COTS stepper controller, the reader is encouraged to obtain the data sheet entitled "Advanced Microstepping Motion Control Chipset, MC1241A", from Performance Motion Devices, Concord, MA, (508) 369-3302. Schematics and PLD listings for this board are also included in Appendix E.

The goal of this design effort is to develop and produce a PCI Bus Stepper Motor Controller PCB for a medical pathology workstation (PCM). The board provides 4 axis (X-Y-Z) motor control of a combination lensless/lensed pathology microscope. Additional high current outputs provide for control of 10x/40x objective switching and linescan loading solenoids. Inputs will be provided for various position and status sensors.

The purpose of the report is to explain the design chosen to implement the intended functions. Other designs may be possible for similar functions, but, in most cases, the reasons why a particular design was chosen over another are beyond the scope of this document.

3.2.3.1 General Topology

3.2.3.1.1 Overall Structure

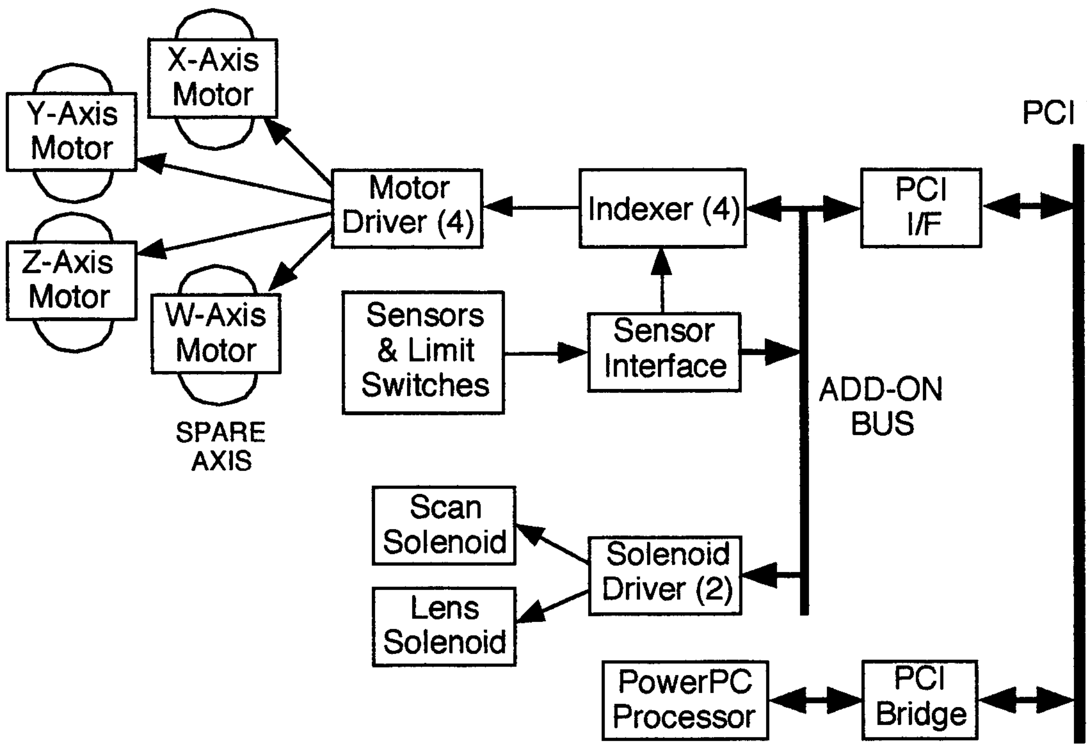


Figure 7 - Overall Structure

A block diagram of the system containing the Stepper Motor Controller PCB is depicted in Figure 7 above. For brevity, a 3 foot long 50 conductor ribbon cable connecting the Motor Controller PCB to the sensors, solenoids and stepper motors within the PCM Optical Assembly is not shown. A summary of each of the blocks follows:

PCI Interface: Interfaces 33 MHz PCI bus of Macintosh 9500 computer to on-board peripherals.

Indexer: Controls ramping (trapezoidal or S-curve acceleration profiles) and provides a high level interface to control a stepper motor axis. Handles commands like "go to position x with maximum acceleration y and maximum velocity z". Indexers may either be microstepping or full stepping. The indexing function is provided by 2 PMD MC1241A chip sets.

Motor Driver: This block contains the current amplifiers required to drive the motor windings. Drivers can be either linear or pulse width modulated (PWM). Linear drivers are simple, but consume a great deal of power if one desires to use the motors near their rated speeds. PWM drives are more complicated but dissipate a fraction of the power. This design uses PWM drivers due to the requirement to dissipate as little heat as possible.

Sensors & Limit Switches: This block contains the limit switches and other sensors (solenoid position detectors).

Sensors Interface: This block contains circuitry to route selected limit switches to the indexer chip sets as well as the capability to read the state of the sensors/limit switches directly from the PCI bus.

Solenoid Driver: A current driver capable of bipolar operation, that is, the ability to force current in either direction through a solenoid's windings.

Scan Solenoid: This solenoid allows the RL4000P fiber optic faceplate to come in contact with the microscope slide for a lensless scan.

Lens Solenoid: This solenoid allows the PowerPC Processor to select the desired objective lens when performing high magnification viewing. A 10x and a 40x objective are available.

PCI Bridge: This is a custom chip developed by Apple called "Bandit" which resides on the Macintosh 9500's main logic board.

PowerPC Processor: A PowerPC 604 RISC processor running at 266 MHz.

3.2.3.2 Detailed Block Descriptions

The following sections describe the above blocks in enhanced detail, enumerating design decisions and IC selections.

3.2.3.2.1 PCI Interface

The AMCC S5933Q PCI Interface IC was chosen to provide the PCI interface to the Stepper Motor Controller board. It is the same IC used on the Frame Capture PCB. On this PCB, only slave mode transfers will be supported, as the throughput requirements are considerably less demanding than the other two boards.

3.2.3.2.2 Indexer

A chipset from Performance Motion Devices (PMD) was chosen to perform the indexing function. The MC1241A chipset consists of two 68-pin PLCC's, and provides advanced microstepping functions for two axis. Since we have three axis to drive, we will require two sets per board. Circuitry to drive a 4th axis will be provided on-board but it's use is not anticipated. Each of the chipsets can generate an interrupt when host action is required. These signals will be "or'ed" together and fed into the AMCC's interrupt input. A register in the Altera IC that houses the Sensors Interface will be readable to indicate the identity of the interrupting chipset.

Microstepped motors tend to operate with much less vibration and noise than non-microstepped ones. Microstepping provides 64 intermediate steps per full step. For a 400 steps/rev

motor, this results in an increased *resolution* of 25,600 steps/rev. The amount of increased *accuracy*, however, is quite less, about 800 to 6,400 steps per revolution, which largely depends on motor construction. Motors constructed with prime ratio laminations offer greatly increased linearity between full steps when driven with a two phase sinusoid. The Semix motors chosen do not have prime ratio laminations, so their absolute accuracy using microstepping is unknown. It may be possible to increase absolute accuracy by calibrating the motors with a high resolution optical encoder. Further details regarding this are beyond the scope of this document. The motors and gearing, however, were chosen to meet specified resolution requirements without the resolution enhancement microstepping provides.

3.2.3.2.3 Motor Driver

The Allegro A3952SLB Full-Bridge PWM Motor Driver IC was chosen to provide the driver function. Each IC can drive one winding bidirectionally ± 2 Amps, so 8 IC's are required to drive 4 motor axis. The driver is capable of two modes of operation: fast decay and slow decay. Fast decay allows motors to run faster at the expense of increased power dissipation. These modes are under the control of an Altera CPLD device which also houses the Sensor Interface. Additionally, a means is provided to transition from microstep to full step statically or dynamically (on the fly) should additional motor torque be required. Finally, the current setpoint for the driver is adjustable from the PowerPC to facilitate a low power hold mode. Power dissipation is a central concern with any high current driver. The PCB is designed in such a way as to dissipate most of it's power through the device's pins to a large area ground plane. The IC has a "bat wing" construction anticipating the need for this kind of PCB construction. A 24 VDC, 2 A external power supply will be required to supply these drivers.

3.2.3.2.4 Sensors Interface

This block receives signals from limit switches and position detectors. It contains combinatorial logic to present the appropriate signals to the limit inputs of the indexer chips. For example, one of the X-axis limits changes depending on the position of the Y-axis in order for the load/unload function to occur.

The board also contains several general purpose inputs and buffered outputs. These are routed into the Altera IC for optimum flexibility.

3.2.3.2.5 Solenoid Driver

This function is also satisfied using the Allegro A3952SLB Full-Bridge PWM Motor Driver IC. The drivers will be configured to use slow decay mode and the current setpoint will be adjustable from the PowerPC or an on-board potentiometer. One Allegro IC will be needed per solenoid.

4. CONCLUSION

The following paragraphs will summarize the strengths and weaknesses of each system. Highlights of what was learned will be discussed as well as suggestions for the next generation workstation. Bear in mind that both are prototypes. Simply put, they work, but the baby's ugly.

4.1 TSS

The TSS was first developed as a rapid prototype system in 1994. New functionality has been added without the benefit of a system redesign of either the hardware or software.

Some of the strengths of the TSS are: (1) it will scan and view an entire microscope slide; (2) areas of interest can be magnified; (3) previous areas of interest can be located precisely; (4) it provides the highest resolution pathology image available in the digital world; (5) it will archive an entire pathology case (the virtual slide and corresponding high magnification images); and (6) most

importantly, an entire microscope slide can be transmitted to another location. The pathologist receiving this image can select areas of interest whose coordinates are transmitted back to the host who captures the images. Those images are then transmitted back to the remote pathologist for diagnosis. The diagnoses and pertinent images are transmitted back to the host for archiving.

The weaknesses of this system are inherent in the rapid prototype development process. First, the system requires a large footprint (takes up a whole desk) and the GUI and software, though stable, are lacking in design elegance.

4.2 PCM

The PCM was developed as a compact version of the TSS. The strength of the PCM is just that, it is more compact.

The weaknesses are primarily a function of the size constraints imposed by the original design. The previous principal investigator desired that the entire unit be housed in a computer tower. That was too compact. Areas to be improved are as follows: (1) illumination system, (2) slide orientation, and (3) focus. See Section 3 for a detailed description.

4.3 Future Work

The TSS can be utilized in a controlled environment for testing. It is inadvisable to fix it anymore. It has proven the concept, which was the goal of this work.

The PCM was a first attempt to develop a commercial prototype. Although plagued with some design flaws, with slight modification it could be a valuable tool to utilize in studies to fine tune the pathologist/machine interface.

If future funding were available, Kensal Corporation would be able to produce a viable commercial product in one more generation. A tremendous amount of information has been learned -- what works, what doesn't. With technology catching up with the invention, this baby will be beautiful!

APPENDIX A

1995 ANNUAL REPORT FOR DAMD17-94-J-4500

TABLE OF CONTENTS

1. INTRODUCTION

This is the first Annual Report for grant DAMD17-94-J-4500 received by our corporation (Kensal Corp.) from the U.S. Army Medical Research Acquisition Agency (Fort Detrick, Maryland). This work is sponsored by the U.S. Advanced Research Projects Agency (Arlington, Virginia) under the Advanced Bioengineering Program. It relates to research and development in the field of medical microscopy, especially for applications in anatomic pathology.

Typically, the anatomic pathologist views, through the microscope, human tissue extracted from about 2,000 patients per year. Ordinarily, this tissue is first fixed in formalin and then, in order to stiffen it before cutting, is embedded in parafin. Slices of tissues are generated by slicing the parafin "block" by means of a microtome to a thickness of about 5 micrometers. Each slice is placed on a microscope slide and stained with a combination of hematoxylin and eosin, biochemical dyes that stain DNA and RNA, respectively. A thin (150 micrometers) coverslip is then affixed in order to protect the tissue sample both from mechanical damage and from oxidation. Thus protected, the stained tissue will last for many years.

Approximately two or three microscope slides are prepared per patient according to the number of pieces of tissue that have been extracted from the patient's body. Each tissue sample is imbedded in a separate parafin block and a slice from each is mounted. (Occasionally, multiple tissue sections may be imbedded in the same block.)

1.1 Report Goals

The purpose of the research described in this report is to (1) become familiar with the working environment in departments of pathology both in the military and civilian sectors in order to plan a field trial of a radically new approach to microscopy for both military and civilian anatomic pathology, (2) prepare a preliminary design of the hardware of this advanced microscope shown in Figure 4 (page 10) of the original proposal, (3) prepare a Software Specification for such a microscope, and at the specific request of ARPA, (4) build in a "rapid prototyping" exercise, using "off-the-shelf" components, two complete microscopes for deployment in an initial field trial before the new microscope becomes available. The goal of the first year was to work on (1) through (4) in parallel.

1.2 Report Summary

This report is in nine sections including this section (Introduction). The Narrative (Section 2) that describes visits to pathologists at both civilian and military hospitals, Hospital Information Systems (Section 4), and Pathology Images and Information Systems (Section 5) all relate to (1) above. The section entitled NeoLensman (Section 6) and Forty Megahertz TC217 Camera (Section 8) both relate to (2) above. Section 7 (Software for PowerMac) treats (3) above and Section 3 (Rapid Prototyping) treats (4) above. The research and development reported in all of these sections (except for Section 3, Rapid Prototyping) has all been done in conjunction with grant 2 R44 GM44420-02A2 from the National Institute of General Medical Science of the National Institutes of Health. This institute has provided funding for an advanced microscope (without rapid prototyping) for the civilian sector. Of course, research and development done under the grant being reported here is directed primarily towards the military sector.

1.2.1 Sections 2, 4, and 5

Research described in these sections (Narrative, Hospital Information Systems, and Pathology Images and Information Systems) was done by Lane Garrett, Jeremy Chambers, and Laurie DeLuca of the Kensal staff. Section 2 is a narrative history of meetings that took place between Kensal staff and pathologists in Arizona. These pathologists were located at hospitals at Luke Air Force Base, Davis Monthan Air Force Base, Mayo Clinic, Veterans Administration, and the University of Arizona. The

goal here was to observe anatomic pathologists in action and to discuss with them the proposed features of the Kensal PCM (PC Microsocpe) which is the acronym given to our advanced microscope. Thus Section 2 consists of a series of trip and visit reports as organized by Mr. Garrett.

1.2.2 Sections 6 and 8

These sections (NeoLensman and Forty Megahertz TC217 Camera) relate to hardware investigations of the TC217-CCD (Charge Couple Device) "chip" that will be used in the advanced microscope. The work was done by Kensal Staff (Shane Chambers and Charles Schoonover) and by an outside consultant (Greg Kline of Kline Research). The TC217 (designed at Texas Instruments, Dallas, and manufactured in Japan) is a remarkable device that permits both dual field and single frame operation according to biases applied by external control circuitry. Kensal is fortunate enough to have in its possession three proprietary cameras that use this device. Thus Section 6 provides a user manual for the Kensal camera. The Annual Report for 1995-1996 will include imagery that specifically relates to this project.

Kline Research, knowing, probably, more about the TC217 chip than the manufacturer, itself, alleged that it would be possible to achieve an operating speed far beyond its advertised specifications. This fact was due that Texas Instruments provides no hardware "drivers" specific to this chip. Therefore, Kline Research was directed to investigate the possibility of building such drivers and operating this chip at 40MHz. The experiment was successful which implies that the PCM camera can operate at design rates that will permit 30 frame per second operation in monochrome and approximately seven frames per second in color (using color sequencing).

1.2.3 Section 7

This section (Software for PowerMac) is an introduction to our work with the Apple Computer PowerPC-chip-based line of processors that have been selected to host PCM. Two PowerMacs (model 9500/120) were purchased. One will be located in San Diego at Ken Crocker Consulting for the purpose of investigating (1) drive circuitry for the microscope stage that moves the microscope slide and (2) interfacing the Kline Research video electronics to the host computer. The other is located at Kensal for use in preliminary telemedicine experiments generating imagery using the TC217 in a "still camera" mode of operation.

Since this host computer is new to Kensal as it uses the PowerPC chip, we requested a consultant (Greg Guerin), who is a PowerPC expert, to begin investigating this computer by running test cases that required conversion from 68xxx code into PowerPC code, using the language C++ as an intermediary. These tests provided us with concrete illustrations of the capabilities of the 9500/120 in data manipulation. Section 7 reports results.

1.2.4 Section 3

This section (Rapid Prototyping) describes the function of the workstation produced under a subgrant to Boeckeler Instruments, Inc. Under this subgrant Boeckeler designed a prototype that could be rapidly constructed using off-the-shelf components. The Nikon Microphot microscope was selected and, in cooperation with the Lockheed Corporation of Sunnyvale, CA, a Lockheed ICON (Image Communications and Operations Node) was procured as the host computer. The Lockheed ICON was modified extensively by replacing its "motherboard," based on the Intel 486, with a dual Pentium computer. In addition a "frame-grabber" was added by purchasing the Matrox Magic card that could receive images from the video camera (a CCIR 601 Sony) attached to the microscope. An ISDN card was also added to permit worldwide communications over the ISDN network. Although two prototypes were scheduled for delivery at the end of the first year, this delivery has now been extended into the first quarter of the second year due to technical difficulties that Boeckeler has had with Matrox and other vendors.

2. NARRATIVE

This section is a chronological file of trips by Kensal staff to interview pathologists and collect information on telepathology and pathology procedure.

2.1 Dr. Joan Hardaway, Davis Monthan AFB Pathology Dept., Tucson, AZ

A 3:30PM meeting on 6/1/95 was held with Dr. Hardaway, with Laurie DeLuca, Victor Carless, Charlie Schoonover, Shane Chambers and the writer in attendance. Dr. Hardaway was a gracious host answering our questions, demonstrating her microscope and showing the lab with its equipment utilized in the preparation of tissue samples, until after 5:00 PM.

Dr. Hardaway does not use any pathology software at this time. But she states that "Copath" is to be made available in the not too distant future. She would like to have access to a database on her computer. Incidentally she just got a PC on her desk and is beginning to learn how to use it with the help of her secretary. Her small work area is void of any additional space. She would like to be able to append data to her data forms, call up the pertinent labs for information when needed, and pull old diagnoses again. Her group is now putting new diagnoses in the secretaries computer thus eliminating recopying by hand. The most savings could be accomplished in the secretaries area since much time is spent in expediting. The hospital uses a wide mix of computers all of which are PC clones-there is no standardization.

2.1.1. Report Generation

In the current pathology system at Davis Monthan, Dr. Hardaway dictates a gross diagnosis of a slide into a micro-cassette, and passes the tape on to the secretary to type up in a report form. Occasionally the pathologists are given the history of the case, but not always, Dr. Hardaway later receives the forms with the gross dictation on it and uses it for referral when she does her microscope examination. When a diagnosis is made, it is typed onto the report form, and in addition, must be repeated onto a card which is stored separately. The reports are stored on the Pathology Department's own Database, and she was not sure if it was tied into the HIS. When asked about the possibility of using voice recognition, she said that it would be more effort than dictating into a cassette.

2.1.2 Peer Review/Referrals

At present it is easy to Fed-X slides for peer review. An on-line connection would save a little time, but the time involved in sending out referrals is not a significant part of the process. She said that more time was spent in doing recuts and packaging. One of the problems in using a courier was that two doctors could not converse about the referral. There is also the problem of a slide occasionally getting lost. Referrals are usually sent to the AFIP. The entire block of tissue is sent to them, which they do not return. Therefore, it is necessary to make any recuts the original pathologists might need before sending the block to the AFIP. Occasionally Davis Monthan uses the Fitzsimmons A-center for their referrals, who do not insist on having the entire block. The number of referrals sent out depends on how many pathologists work together. If a pathologist works alone, a lot of referrals are sent out, even if he/she is certain of the diagnosis. This is done for documentation, risk, and liability purposes. If a number of pathologists work together, they may consult with each other in place of sending out for referrals. About 10% of her cases are sent out to another specialist for liability purposes, of these, most are skin tissue.

If an image is of the type that could go on line and get 24 hour response, "It alone could help a lot". If it could be done real time it would be "great". However she stated that we need to improve the Digital Microscope. Most of her work is done at 4x and 40x and occasionally at 100x with slide-oil-objective for special cases such as bone marrow (less than 5% of the time). The

repeatability of slides (exact same spot being under the objective) is very good. Even with 100x and oil, repeatability is satisfactory, assuming the same microscope settings. The size of slides is very uniform from unit to unit, but she did not have any specifications as to tolerances.

2.1.3 Digital Advantages

Digital capabilities would be great to mark a spot or Area Of Interest (AOI). In some cases such as individual cells a Cytotech checks for all cells, and makes a circle around each bad cell. The Pathologist then looks in the circles for diagnosis. ." According to Dr. Hardaway, cytology requires lots of searching time. She suggested that a digital microscope would help by finding pre-marked regions of interest. There is now some equipment using Neural Nets that picks out enlarged Nuclei and marks AOIs. This is a case where the computer is better, this is too tedious and a human can miss some areas.

2.1.4 Case Load

Cases of a given type seem to come in cycles, She ran about 128 Gynecology slides in the first 5 months this year with up to 5 slides per case. Non-Gynecological slides such as urine samples were about 75 in the same period with about 2 slides per case. She also does Biopsy slides which are cyclical, especially for possible skin cancer. A typical day's work consists of 1 to 3 trays of 18 slides each.

2.1.5 Processing a Glass Slide

On an average day, Dr. Hardaway receives a cassette of gross tissue that has been assigned a number and entered into a log book by a technician. She views the tissue (in the cassette) and describes what she sees. She then takes her own sample of the tissue to be placed on a glass slide. The process of taking a sample involves putting the cassette (that has the tissue in paraffin) into a heated processor overnight. This allows the technicians to properly embed the tissue sample. Next the tissue is cut into thin "ribbons" and floated on a water bath. A glass slide is immersed in the water. placed under the tissue, and lifted out so that the tissue adheres to the glass slide. The slide is then "cooked" to set the tissue in place, stained, and has a cover-slip placed upon it. when the process is finished, Dr. Hardaway receives a tray of slides, all of the same tissue contained in the cassette. She explained that for some cases, she must have several (three or more) layers of the same tissue to observe, to be sure of her findings. If she were given one or two layers, certain details might not be present that are key to the diagnosis.

2.1.6 Record Keeping

About one years written history is kept locally, (on 4" x 6" cards). Slides are stored in a local slide library. Bar codes are OK for samples in the lab, for tubes of blood, etc. but not for slides.

Getting reports to clinicians is the biggest bottleneck in the system. There is at least a 24 hour turnaround time. She does proof-reading of the transcripts dictated on tape to her secretary. The dictation machine and transcriber can sometimes present problems. They can now sign once on 4-ply paper forms - which saves some time. Dr. Hardaway stated that, ideally it would be helpful to be able to pull up information from another report, but finding a full report is unlikely. Accessing on-line records would replace the running around to fetch files by hand, and would save the redundancy of typing up the diagnosis onto a separate card. It would eliminate the "inefficiency" delays that occur. The pathology module takes up most of the secretaries time, but not the pathologist's time. Dr. Hardaway's major constraint is the budget.

2.1.7 Additional documentation

Dr. Hardaway gave us a Tissue Examination report form and a flyer on the Computer Trust Corporation's SURGE™ Single-User Core Module: inexpensive software for the smaller Anatomic Pathology Laboratory. the core module is $3K with 8 additional modules at $1K or $2K each for a total package of $14K and a $2.5K yearly maintenance thereafter. The package does not even run under "Windows".

2.1.8 Conclusion

As far as improving time and physical motions, Dr. Hardaway said that nothing could beat the microscope. Also, "The microscope image cannot be beat." According to Dr. Hardaway, pathologists are very comfortable with the current microscope technology, and she insists that a digital microscope <u>would not</u> replace an optical microscope. She does, however, like the idea of a digital microscope for assisted diagnosis. A Pathology workstation with digital features could be a useful tool to assist in the process. It would be used in special cases for indexing the slide, marking and returning to AOIs and possibly for data and image storage and retrieval. Sometimes slides are lost.

2.2 Boeckeler Instruments, Inc., Tucson, AZ

A 2:00P meeting on 6/22/95 (scheduled last week) was held with Steve Lange, and Bill Berchard of Boeckeler, with Victor Carless, Jeremy Chambers, Shane Chambers, Laurie DeLuca, Charles Schoonover and the writer from Kensal Corp. The purpose of the meeting was multifold, to answer Boeckeler's request for an update on what the Kensal PCM team were doing, Introduce new team members and their areas of responsibility, facilitate interaction and cross-fertilization, observe the microscopes and dual Pentium microprocessors, and ascertain Boeckeler's progress in both the software and hardware areas.

2.2.1 Hardware

Both Nikon microscopes are in house at Boeckeler, but the second unit is not compatible with the dual Pentium PC. Although ordered and shipped at the same time, the second one has different electronics in the base. They are in the process of getting this resolved with Nikon. The main problem is still the integration of the Kodak chip and the Matrox board. The Kodak chip has been upgraded to 8Mhz clock rate with a new EPROM and crystal, which should have solved the problem. The chip output looks OK on the oscilloscope, however it still is not working with the 3 megabyte Matrox frame grabber board! Boeckeler is still having communication and documentation problems with Matrox. They finally got the ISDN tariff approved, but do not have a firm install date. The ISDN link is pretty well handled by the NT operating system.

2.2.2 Software

The Software effort is mostly complete with The Front End and major Objects coded. About 2,000 lines of original code have been written with the total probably over 25,000 lines considering overhead and library items. A couple pieces of code will have to be written for the interfacing of the ISDN to Windows NT (a minor effort). Also the drivers for the microscope stage are not completed. They will interface with a new stage driver board which powers DC servos with encoder feedback. Bill offered to give our team a 3 hour briefing (evening preferably) on his software specifications, flowcharts, objects etc. I will try to set up a meeting in a couple of weeks.

2.2.3 General Miscellaneous Information

Boeckeler now has about 75 distributors (primarily microscope distributors in the Electronics and Industrial markets) with about 15 to 20 of them international. LEAD Technologies, Inc.; 900 Baxter ST; Charlotte, NC 287204; (704) 332 5532; FAX (704) 372 8161 has done a good job on their software toolbox for GUI and Image processing for Windows. Their 4096 X 2116 X 24 bits depth guide image takes about 26 Megabytes of storage. The Matrox frame grabber handles 1024 X 768 pixels. We found out from Steve Lange that the Department of Pathology, U of A have found that the Roche system looks worse at a resolution of 3200 pixels than 1000 pixels. Their current work is primarily done at 1000 pixels. The Microscopes have a Sony 3-color high resolution camera (768 X484) with on top controls for the adjustment of all key parameters. We all agreed that real-time focus was a great feature. The Sony images looked very good, giving a good impression for an instrument cost of about $6K. The Matrox board came with a bundled Pixel Viewer (Frame Buffer Viewer) that appears quite useful. Latest information indicates that they will be able to scan in rearrange and complete a full guide image in about 20 seconds. Additional server software will be required in the future if we wish to include a third active station.

2.2.4 Conclusion

Hardware, software, and ISDN hopefully will all come together for the start of peer to peer testing by the end of August. This looks like an overall slippage of six week to me.

2.3 Dr. James Byers,VA Medical Center, Tucson AZ 85723

On June 28th Jeremy Chambers, Shane Chambers, Laurie DeLuca, Charles Schoonover , and the writer visited with Dr. James Byers at the VA Medical Center in Tucson. Objectives were to learn about his procedures, methodology, and computer and Hospital Information System interface if any. Another major objective was to discuss telepathology to the extent that he was familiar with advantages and disadvantages of the current technology.

Dr. Byers was interested in our activities and was even familiar with Arvie's Work at Boeckeler (he is a neighbor of Arvie's). A gracious host and good educator, he gave us almost two hours of his time. Our group of five from Kensal was directed by Dr. Byers to small room adjacent to a clerical pathology office. Dr. Byers manned the main microscope which had four other Binocular stations for other viewers. Two cylindrical extensions in opposite directions from the main microscope carried the light approximately 2 feet from the source to the small-footprint observer stations. The observer mounts could each swivel approximately 90 degrees about the normal to the desk-top plane and were positioned opposite to each other. Dr. Byers could point to regions of interest within his field of view with a super-imposed transparent arrow. All other viewers could also see this arrow. The main microscope had several levels of magnification and illumination. A Sony 3 CCD RGB camera was mounted above the main microscope. The camera's real-time output was directed to a high-resolution RGB monitor and a Polaroid digital film recorder for color hard copy output.

2.3.1 Case Analysis

Dr. Byers explained that pathologists may see several types of specimens per day. We discussed types of cases that did not require structural analysis such as blood cell examination. One type of specimen he referred to was called a peripheral blood smear (usually from the Hematology Dept.). This is a relatively simple procedure, commonly performed in doctor's offices, in which a needle is placed into a peripheral vein to draw out some blood. When pathologists receive a peripheral blood smear, There are basically three types of blood cells, red cells (about 7 microns in diameter) which are often used as a reference for relative cell size, and

two basic types of white cells. Particular attention is given while viewing the slide, looking for any inconsistencies.

A second type of specimen a pathologist might receive, is a cytology study. These are typically Pap Smears in which the pathologist is looking for cancer. Smears, such as Pap smears do not have any tissue structure. Often Cytotechs will be used to examine the whole area of the slide and mark any suspicious ROI's, thus saving time for the Pathologist.

In similar cases Radiologists may suspect a lesion (such as found in the Lung) and request a biopsy. This frequently will be an aspiration which is almost non-invasive but usually results with enough material for a satisfactory diagnosis. With a Fluoroscope the Doctor can accurately observe the position of the needle and get a sample from the area of interest. A Valley Fever case was used to illustrate the point. The pathologist will then view the slide to determine if a biopsy is necessary. If it is not necessary, the physician and patient can avoid an invasive procedure all together. The role of the pathologist here and in other procedures is to make available as much information as possible to both the clinical physician and the patient. More information can be determined from tissue than from any other procedure.

In many of these "non-structure" cases only one or two slides are necessary. (Dr. Byers at this point let us view a Cytology Slide of a Pap Smear. He explained that he was looking for clusters of cells, of which there were not many.) The examination of a case can take a few minutes, especially if a Pathologist is confirming a prior diagnosis or up to an hour on a difficult case where he may also go back for more samples.

For cases where whole tissue is sampled, structure is important. When Dr. Byers looks at a slide under the microscope, he looks for the pattern of growth, the array of cells, and the architecture when trying to make a diagnosis. He does not just look at one cell or one cluster of cells. Dr. Byers often cycled the microscopic focal plane through most of the tissue regions he was examining under high magnification. This appeared to give him further structural information about the specimen i.e. the thickness and location of object-boundaries in his regions of interest. He usually looks for two or three other cells or features to confirm what he is thinking. It is crucial that Pathologists be accurate in their diagnoses. The difficult cases make up less than 5% of the total cases.

Cancer cells often have more than two chromosomes and appear more mitosis than normal cells. This is why they usually show darker than normal cells (the stain preferentially shows chromosome material). In unusual cases special stains may be ordered to show special immunological effects. Use of antibodies which react with cytoplasm is being utilized more frequently. A difficult case may involve an area where the pathologist has not had much experience, or a rare type where referral to expert(s) is required. Referrals are usually to AFIP, however a pathologist is free to go to another preferred source for a particular case.

In discussing briefly the role of Cytotechs, Dr. Byers said that they were limited in how many slides they could view in one day. The limitation was placed on them since fatigue can set in and result in sloppy work. When the Cytotechs at the V.A. Hospital find an abnormal cell, they mark a blue dot in the 9 o'clock position adjacent to the cell so that the doctors know where to look. Other cytotechs at other institutions have the equipment to place a ring around the bad cell. Quality control could probably be enhanced in some cases where "rectilinear or rigid" scanning swaths over the specimen are used during the initial analysis. Rectilinear/rigid scanning refers to using strictly orthogonal stage motion where either the x or the y (but not both) motion control is used independently. Thus, the specimen is viewed at a fixed magnification starting from the left end (and proceeds to the right end) of the slide and is completely studied by moving through a snake-like path of overlapped linear scanning swaths. This is a very tedious process and is usually performed by a technician. The cytotech takes a rectilinear look at a case, scanning the whole slide.

For tissue where structure is important, a Pathologist may take multiple cuts and also look at the structure in "3D". Each slice cut with a microtome ranges in thickness, about 3 to 5 micron per sample. If the slicing is too thick, pathologists can't see all of the morphology to make a proper diagnosis. Dr. Byers explained that there is no universal convention for slicing tissue, that it depends on the institution where a pathologist works. The slicing procedure is important though (which direction it is sliced in, etc.), and this is why they perform a gross examination prior to the slicing. A gross exam reveals such thing as the mobility and color of the tissue to the pathologist. This type of information will determine how the slicing is to be done. An example would be Renal (Kidney) tissue.

Biopsies of skin lesions are sent in by regular doctors who give a description of the case along with patient history for diagnosis by the Pathologist. Tissue samples of this type or those from tumor biopsies usually require extensive structural examination. Here a high resolution guide image would be required. (As a point of reference 80% to 90% of the better images available on-line are satisfactory for diagnosis.) For the more difficult cases a "Differentiated" Diagnosis is performed with possibilities arranged in decreasing order of probability. Additional tests are performed, samples taken, peer review requested etc., with exclusion or confirmation until the extra options are eliminated. Some cases are just "tweeners" where the case seems to fall between two categories. Where structure is important, several slides per case are the norm with difficult cases often taking more.

2.3.2 Telepathology Comments and Observations.

Dr. Byers is not hooked up to a telepathology system, however his demonstration/conferencing area with the five-station binocular microscope gives some of the same features and is used primarily for training. This is almost like an internal telepathology setup. The main microscope is equipped with a Sony real time camera hooked up to a Trinitron monitor. There is a video link to an upstairs operating room that is not normally used.

Dr. Byers focused on demonstrating pathological methodology and analysis of stained slide specimens from several case studies. There is no standardized method that pathologists use to measure the accuracy or quality of their diagnosis over time.

KSC asked if having knowledge of the percentage of total tissue viewed could be used as a gauge for quality control. Rather than answer directly, Dr. Byers demonstrated several scenarios where such knowledge could be useful and where it may not be useful.

Dr. Byers said that older pathologists are much more skeptical about telepathology than younger pathologists. (His mind-set is with the younger pathologists) The younger doctors are growing up with computers, etc. and are therefore more prone to use telepathology. Before Dr. Byers transferred over to the V.A. Hospital, he went to China with other pathologists from UMC to work on a telepathology system. They used telephone lines for transferring images, and typically transferred a case (5-7 images) at one time. It took anywhere from 30 seconds to 5 minutes to transfer each image at 14,400 bps depending on the phone lines, etc.. One problem that he encountered with the images was the low resolution. A second problem that he predicts, is that real-time consultation and data transfer will not be reasonable to use in the real world of Pathology. He explained that the expert on the other end of the system is not going to want to sit around and wait all day for a Pathologist or clinician to transfer images at his convenience. He said that appointed times are a must. Images and comments can be transferred ahead of time without problems. Dr. Byers does feel that telepathology can be used, though, if properly applied.

Dr. Byers feels that low power magnification for guide images is still somewhat of a problem in that resolution is still to limited, however the best available over the Web are usually

acceptable. He foresees a Quality Pathology Workstation, where text, arrows, verbal comments could be added to high resolution images for transmission to an expert center such as Memorial Sloan-Kettering Cancer Center. For example when 5 cases and their images are up-loaded, the experts could be called for consultation. Perhaps sessions could be scheduled every 10:00 AM to 10:30 AM on Monday, Wednesday, and Friday. On-line real time consultation is probably not good unless it is with a very remote site with no pathologist, maybe with a cytotech or regular physician. Some areas of diagnosis will be more of a problem such as skin lesions where 80% to 90% are hard to determine and Lymphomas are always a problem. The UoA seems to be having some success with images regularly coming in from Kingman, AZ and Mexico for example.

2.3.3 General and Miscellaneous Information

Dr. Byers explained that pathology has been very descriptive for the past 150 years (definitive categorization of diseases). Descriptive criteria often includes how much blue there is from the stain (the more blue, the more DNA), how big the nuclei are, and the amount of mitosis going on (which determines the rate of growth). Pathologists have been adding to the database of "Histopathologies"? since 1860. Pathologists get as much information as they can and will now even give a Prognosis in cancer cases where a person has a probability of living for a given length of time. Pathology is currently moving toward more focus on the genomic structures of cells (DNA make-up, etc.). In addition, more usage of antibodies which react with cytoplasm is being promoted.

A Cytotech will have a BA and at least one year of training while a Pathologists requires 10 years of training to become an expert in a specific area. Cytology studies the patterns and numbers of cells in a sample. "Pink" and "Blue" are the colors resulting from the standard stains. Experienced pathologists normally do not use atlases for their areas of expertise, but may look up information on unusual cases or cases in areas outside of their normal area(s). In a timed mock run of using an atlas for reference, Dr. Byers found what he was looking for in approximately two minutes. He stated that they may use an atlas for referencing about once a month. A specific "look-up" may only take a couple of minutes, however the atlas had generally poor photographs. (**Diagnostic Surgical Pathology**, 2 volumes, Editor: Stephen S. Sternberg, Raven Press, ISBN: 0-88167-442-7). Many pictures were black and white, a few were color but all were of relatively poor resolution.

QA processes are coming to play in the Pathologists' world. Photometrics are being developed and peer review is being encouraged. This is currently low key at the VA hospital, however cancer cases are always reviewed by two pathologists "independently".

The CAP is now accrediting Laboratories. More frequently pathology labs are seeking accreditation with CAP or other state agencies to improve their image and help standardize their operations (which we assume makes referrals easier). CAP accreditation requires that certain methodologies (not described) be adopted and reports (not described) be made regularly through hospital management. Dr. Byers said that at first the standardization of operations was welcomed by pathologists but that there has been some resentment toward CAP for over-policing (no further explanation was given) their work. The paperwork and additional checks and balances add costs but do add to the status or prestige of the hospital. Administration does keep track of the work load which justifies staff and expenses. Naturally many reports are generated for the VA. KSC asked Byers about his interaction with top management in the hospital. He said that reports describing work load and throughput (i.e., patients, slides, paraffin blocks, etc.) were compiled monthly on paper and given to management. Byers did not know any information about the VA hospital's HIS or whether his secretaries had access through their terminals in the adjacent office. Our impression is that Dr. Byers likes to keep his distance from the HIS world. He seems to leave this up to the secretaries.

Dr. Byers is connected to internet and the Web at 14.4Kbps through PrimeNet, a local service provider. The hospital is planning to have direct access in September. He shared with Laurie his list of the best Web sites for good pathological images that he had previously compiled. He observed that the JPEG images are of better quality than the GIF images. Laser Discs usually do not have acceptable images. He plans to use the images primarily for teaching.

One of his associates (who was not in at the time) is their "Local Hacker Pathologist (forgive the pun)", **Ron B. Schifman**, (MD, Staff Pathologist). Ron is hooked up to the local Windows net, has a Sony monitor (estimated 640 by 480 resolution), a 1 gigabyte hard drive, a five CD ROM drive with random search, and a color printer for recording images. Ron is planning to get a better resolution color printer to improve his output. There is also a RGB camera and microscope hooked up to their local net. He has made "Medline" CD-ROMs accessible to all networked staff. "Scientific American Medicine" CD-ROM is planned next. Bibliographic retrieval is a hot item for the new medical staff.

2.3.4 Suggestions for a Pathology Work Station

Some features to include would be specialized information helpful in the diagnoses of Hepatitis Types A, B, or C. It would be nice to pull up the patient enzyme test results along with any available expert data base information. At first look, a Pathologist may not refer to the Physicians data and patient history thus giving an unbiased view, since the clinician often states what he thinks the disease is. Having this data on-line could be useful for later reference. In reports, SNOMED is often used along with some Pathological abbreviations such as BCC for Basal Cell Carcinoma. Alphanumerics are used to help cut time on about 90% of his cases. Having some reference books on line such as Sternbergs and Robbins? could also be useful.

2.3.5 Conclusion

Dr. Byers personally has archived 16,000 Kodachrome slides of his work. Perhaps a digital archiving system could come close to the costs of film and provide more convenience and quicker access. With improvement the PC Microscope and Telemedicine will be useful in a variety of cases. "Better" guide images, on-line references, annotation capabilities, and quick archival ability, would add to the PCM utility.

Dr. Byers was most helpful, an excellent host and educator with a real interest in using the latest technology wherever applicable.

2.4 Review "Windows" Software for Boeckeler "Off-the-Shelf" PCM

On Wednesday the 12th of July we have scheduled a visit with Bill Berchard of Boeckeler Instruments and Jay Nance at KSI (the Executive Conference Room). The purpose of the meeting is to discuss and review Boeckeler's "Windows" software as used on their "off-the-shelf" PC Microscope design.

We will meet at approximately 6:00 PM in the large Executive Conference room. Due to the hour, and Jay coming down directly from a course in Mesa, Pizza and soft drinks will be served. Bill has a broad software background with IBM and a lot of experience with "C/C^{++}" on the PC platform. His approach, documentation methods, and software specifications for the dual "Pentium" workstation will be educational and most helpful toward our efforts. We will soon be starting the PC Microscope projects where we are striving for multiplatform capability. Jay is currently completing Kensal's "C/C^{++}" software standards and will be helping us with our software specification and coding efforts on the PC Microscope. The meeting is expected to last until 9:00 to 9:30 PM.

2.5 A.K. Bhattacharyya, MD, UofA, College of Medicine, Tucson, AZ

A 9:55 AM meeting on 7/19/95 was held with Dr. Bhattacharyya, Laurie DeLuca, Victor Carless, Jeremy Chambers, Shane Chambers, and the writer in attendance. After briefly previewing the planed meeting, Dr. Bhattacharyya led the discussion and demonstrations until 11:15 AM. He was very responsive and open to our questions and only begged ignorance on some of the technically oriented queries. Data gained from the conversations are enumerated in the following paragraphs but not necessarily in sequence. Dr. Weinstein and Dr. Martinez were unable to attend. We will try to meet with DR Martinez at a later time for some of the more technical system questions.

Dr. Bhattacharyya explained that the pathology group at UMC specializes in OBGYN, liver, lung, kidney, and hemopathology cases. All Doctors have their subspecialitie(s). The cases get reviewed by three pathologists to ensure the quality of the diagnosis. It was not clear as to whether all of these organ systems are being reviewed on a regular basis with the Roche telepathology system. Only the "questionable" cases are being referred to them and about two thirds of these can be diagnosed satisfactorily with the system. Of the remaining cases additional information, tissue samples, or other regions of interest are requested and about half (one sixth of the total) require the glass slides. The doctor doing the diagnosis is dependent on the remote Dr. for selecting and sending the proper images, this is not perceived to be a problem. Case information is transmitted with each set of images. A report goes back to the referring institution. About 400 telepathology cases have been done to date. The system is useful in Triage cases. Dr. Bhattacharyya spends an average of 7 hours per week (10% of his time) in telepathology. The telepathology system with improvements can be useful for Morphometrics, Cytology, Consultation Services live teleconferencing and in the future providing and archiving size and dimensions of ROI's. For Quality Control purposes they are getting 4 to 5 consultants to review each of their cases.

The Roche system used NEC Multisync 17" monitors with an apparent .28 dot pitch and an estimated refresh rate of 60 Hz due to the noticeable flicker. The Roche telepathology computer system was based on a 80486 with 32 MB Ram, a 24-bit graphics card (NDI TIGA True Color?), MS Windows OS, a high-speed modem (unknown bps), and some telecommunication and graphics software (called ImageManager). We saw two of these systems that were installed at UMC, similar systems were installed at each referring institute.

The telecommunications software is constantly left in the "auto answer" mode so that it may receive images without human supervision. They are using the new version (2.2) of the Roche software. An Olympus Camera was mounted on top of the microscope in their upstairs computer laboratory which was in turn connected to another Roche computer. Images received by the system seemed to be a constant 1024x774 pixels in size (24 bits-per-pixel). Image transmission for this size image took 2.3 minutes per image, actually 7 minutes for three images. Image quality and contrast varied with the Doctor who set up the sending system and his experience level. The best images come from Dr. DeLeon in Mexico. He is the chief of pathology in Hermosillo, Mexico. The paint-to-screen of the 1024x774 image from its decompressed archive was slow (about one or two seconds) on top of the slow decompression time (about three seconds). It took about one minute and 20 seconds to find and pull up old images from the archives. For demonstration 6 images were retrieved. Total load time can take up to 20 seconds. Other image sizes apparently are not used since the monitor can not resolve them. Remote sites can scan-in images of 1536/1160, 2044/1450, and 3072/2320

Dr. Bhattacharyya called Dr. Fleishman in Cottonwood, Arizona at 10:05 am but was unable to get him. He then called Dr. Nelson in Kingman and asked him to send several "colorful" images to us for demonstration purposes. There was a delay of about 15 minutes before the

transmission started. Two plain push button phones were available near the Roche computer for such calls. Dr. Bhattacharyya said that he preferred a hands-free conferencing/speakerphone for collaborative specimen discussions with remote groups. Dr. Bhattacharyya struggled to dial Dr. Fleishman and Dr. Nelson--he looked around backwards to the computer screen for the number and then turned to the phone to type a few numbers and repeated this action. An auto dialer built into the software with telephone conferencing ability could greatly simplify these manual operations. Dr. Bhattacharyya had difficulties in navigating the software. Any software we write must be as intuitive as possible, and any non-trivial operations should be hidden away so only a technician can get at them.

Dr. Nelson's case was a needle biopsy of a tumor that immediately looked like Breast or Prostate tissue. No information was initially given to prevent any bias. Three images were received 4x, 10x, 40x. All three images were looked at simultaneously on the screen. A diagnosis of High grade Prostate Cancer was completed in 3 minutes. (Dr. Nelson also had a 60x lens available on his microscope.)

DR Bhattacharyya could overlay red (or other color) arrows and outlines of any orientation along with text on top of an image. Further, outlines could be made by point- click-drag operations. Text can also be annotated to the image. There is also the possibility to call up a history file that can be transmitted with the image. The overlay could later be removed to reveal the underlying imagery. A case from Mexico was shown with an excellent image of a Subcutaneous Lesion. There is also the ability to scale images, scroll them up and down and tile multiple images on the screen. Thumbnail images are used to give a quick look, on screen size is estimated at 128x128 pixels. The software package has a generally good looking GUI with pull down menus, buttons for action selection, and scroll bars where applicable. The intuitiveness and logical progression of the system could be significantly improved. Dr. Bhattacharyya had not used the system for two weeks and had to rediscover where to find some items and get the desired information. Dr. Bhattacharyya wished there were *morphometric* related tools in the software so that he could compute area, length, etc. of the outlines or other markings. He said that Roche is "looking into" adding such features to the software.

Maximum storage in the system is 500 images. The images are then archived by Dr. Martinez, a University of Arizona Electrical Engineer. All image file names appeared to be based on the date they were received and were placed into a single directory. All image information is manually logged into a blue ring binder for later reference. A relational database like "FoxPro" for archiving and retrieval would be most useful.

Dr. Bhattacharyya repeatedly had to reload the imaging software because of mistaking the "File:exit" command with other functions. Reloading the software took roughly 50 seconds. Dr. Bhattacharyya allowed Shane Chambers to work with the operating system on both of the Roche computers to determine the system makeup. Dr. Battachara, proclaiming his computer illiteracy, marveled at Shane's ability and suggested that he be hired to operate the Telepathology system instead.

The original cost for the hardware was $25,000; $3000.00 for the microscope, $7000.00 for the camera and $15,000.00 for the computer, etc. Additionally the software cost $15,000. UMC thus paid a total of $40,000 for the system. Specifically how much was spent on the two Roche 486 computers is unknown. Three software updates have been received since the system has been in operation. The system in Dr. Weinstein's has had a camera change from a German to a Sony camera. An Olympus microscope is used. The focus feature (maximize a contrast number on the screen) was demonstrated. The ocular and screen focus were different, requiring final adjustment by the system optimization routine. A few of attempts were made with an excellent image finally obtained. Focus is adjusted in real time on a thumbnail view of about 128 x 128

pixels estimated. Time to adjust and capture a good image is about two minutes (not counting relearning the system).

Dr. Bhattacharyya recommended that we speak to Dr. Martinez (621-6174, Secretary: 621-2718), who was unavailable during our visit, for technical questions and a discussion of UMC's HIS.

Within Arizona, five hospitals belong to the triage telemedicine network (UMC is the hub for these hospitals). U of A also networks with Dr. Zimmerman in Mesa and Dr. Alverez in Yuma. They have had a few trial runs with a hospital in Sedona. Dr. Bhattacharyya commented that thus far the network has been limited to just state-wide hospitals since legal issues for inter-state networking have not yet been fully resolved. Some progress has recently been made in Congress however. Test cases have been run "unofficially" with a hospital in Colorado that would like to be in the network when possible. Approximately 20 phone numbers are listed in his "phonebook". International connectivity has been established between China and Mexico with good results. To date, 400 telemedicine cases have been handled by Dr. Battachara and the group. We thanked Dr. Bhattacharyya for the more than two hours of time given to our group.

2.5.1 Conclusion

Telepathology regardless of complaints and objections, is frequently a useful tool that with, changes in legislation, improved hardware, better user-friendly software, and education to create more familiarity among pathologists, can be a productivity tool which also enhances the quality of diagnosis through team effort. Being able to call-up old images cases etc. by a relational data search would be most useful. The Roche system does not have an archiving feature. Frequently Dr. Bhattacharyya will set up a 35mm camera on a tripod in front of the screen to capture images for his records. This costs $8.00 an image instead of an "$800.00" charge to send out and have images transferred to film. Archiving is even more important in training hospitals where certain illustrative images need to be handled by multiple individuals. Dr. Bhattacharyya briefly mentioned the possibility of image enhancement and quantitative DNA analysis for special cases e.g. (Pap smears). Roche indicated that they are "looking" into this area. This could be a subject for our future study.

2.6 Scottsdale Memorial Hospital

AM and PM meetings were held with M. Pattie Madjidi regarding HIS, and Dr. Peter Jolma regarding pathology respectively. Scottsdale Memorial is a progressive hospital and willing to try the latest developments. They evaluated the latest HMO HIS integrated system with an advanced GUI and turned it down due to its very slow operating speed. The old DOS system was significantly faster. The main computer storage database is housed at Scottsdale Memorial on Osborne Road (SMO), while most of the pathology work is done at Scottsdale Memorial North (SMN) off Shea Boulevard. SMN works closely with the pathologists at Mayo Clinic. There will be less interaction when Mayo builds their own hospital north of Bell Road near Scottsdale Road.

2.6.1 CoPath

Pattie Madjidi, MT has broad pathology knowledge and administers the CoPath system at Scottsdale Memorial. She was most helpful and kindly responded to all our questions. The CoPath installation and its linkages with other systems at SMN and SMO was the center of our discussion.

CoPath is a complete stand alone system. It was chosen by Scottsdale Memorial North because it is the "Cadillac of pathology systems, and very easy to use," Ms. Madjidi said.

HBO&C's system was much harder to work with; it had a very poor user interface and used a lot of screening.

The way the CoPath system works is that everything must be linked through Word Perfect 5.1. This limits viewing capabilities into other areas, such as lab reports, because since CoPath is not Windows based, the system must be exited before entering into another one. When Ms. Madjidi performed a demonstration of this, the transition took about 20-30 seconds. Her one improvement to the system would be to integrate into the pathology system so that there would be an easier way of viewing other areas without exiting the CoPath system. She said that it would be nice to have a patient demographic link to the HIS.

When asked if the physicians used the computers, Ms. Madjidi said that they use charts instead of computers for recording the history of patients because it was much quicker for them.

CoPath was installed at SMO in 1991; since then, no old records stored in the CoPath system have had to go off line into storage--plenty of space within the system is still available.

Pattie explained that it was hospital policy to purchase only HP PC's. CoPath was installed on these HP PC's. However, CoPath exclusively deals with Dell PC's, thus the $11,000+/yr. CoPath service contract with SMN did not cover PC hardware problems and was limited to maintaining a bilateral interface between a 486-based file server (with an oddly 14MB of RAM) for CoPath and three "ARNet Panels", apparently some kind of interface hub for all the networked PC's and printers using CoPath. Collaborative Medical Systems (CoMed) is responsible for repairing the software of the system. When a repair is needed, it is done over a modem. Scottsdale Memorial is a Hewlett-Packard user, which is non standardized equipment for CoMed. This is why they are only responsible for software repair. In addition, CoMed is only responsible for getting information from the file server to the RNet ports. Three RNet panels are used as the ports in the system. From there, the hospital uses in-house hardware for the rest of the connections. All of this equipment resides at the Osborne location of Scottsdale Memorial. It has a real-time link with the other hospital. CoPath maintenance is always performed through a modem interface and is done remotely. SMN employs a technical crew on site to maintain the PC hardware, among other unspecified duties. CoPath at SMN was linked to SMO via a T1 line. The CoPath file server is located at SMO.

Installed on the HP's along with CoPath were: WordPerfect 5.1 (word-processor, not the latest version available: 6.0), Software-Carousel (an old, out of production, DOS application that allows several applications to reside in memory simultaneously, with hot-key inter-application switching), and a terminal emulator. Neither CoPath, nor these other applications were Windows-based; rather all four were DOS-based. CoPath Windows-based workstations are available but Dr. Madjidi said there has been no effort put forth yet to acquire these.

Pattie Madjidi operated through the CoPath user interface in a very fluid manner, and appeared to know well all the features available in the CoPath system. CoPath appears to have both developed a very natural user interface for Pathologists and provided the appropriate end-user training.

When asked what the weak points of the CoPath system were, Ms. Madjidi stated that there really were none for her. She likes the system and said that it runs great. She did show a preference for a Windows environment, however. In addition, she tends to like the idea of a touch screen, although no one else has shown an interest in that. She stated that CoPath is the "Cadillac of Pathology Systems" as opposed to HBO's pathology module, which was more "cumbersome" to use. Recall, CoPath went live in 1991 at SMO/SMN and that all records entered since then are on-line. We attempted to clock the time required to call up records since 1991 through 1995, however, retrieval was almost instantaneous for records as old as 1991. Recall that the file-server

for CoPath was located remotely at SMO, some 16 miles away, available through a T1 link. The CoPath system is MUMPS based.

The term *accessioning* refers to assigning a number to a specimen case. A patient may have more than one case, and if that is so, their date of birth or name regulates all of them together. Accessioning numbers are regulated by having a prefix of some sort (for example, "N" for the north hospital) at the beginning, the year it was produced next (i.e. 95), and then the case number (the sequence of how the cases come in). There are then an unlimited number of parts that can go under each case. For example, a slide may be labeled with Part A, which refers to the liver. A second slide may be labeled with Part B, which refers to the lung. Finally, there is a block listed under the part or subparts. The blocks refer to the different areas taken from a chunk of the tissue from the paraffin block. The number of blocks varies depending on how many areas the pathologist wants to view. The CoPath system prints labels like this for the slides produced at the hospital. The information must first be entered into the system, and is usually done so by a surgical pathology assistant instead of the pathologist. It should be noted that multiple slides are made per block, based on how many different types of stains are to be done, and how many slides the pathologists wants from that particular block.

Accessioning, or the assigning of alpha-numeric codes to pathology cases, is furnished automatically through CoPath and is based on a coding-system designed by Scottsdale Memorial. Accession labels are printed in batch and attached to all bags of paraffin bounded specimen blocks and the resulting slide-specimens. A typical accession number is as follows:

N95-1411

The "N" is an origination indicator which stands for Scottsdale Memorial "North" Hospital. The "95" is the origination date (1995). The 1411 is the case number.

The alpha-numeric hierarchy in the accessioning process is illustrated (extrapolated from Pattie's description) as follows:

```
N95-1411 : SMN 1995 case #1411
      |
      +-->N95-1411 A : Liver
  |
  +-->N95-1411 B : Pancreas
              |
              +-->N95-1411 B1 : Pancreas, block 1
              |
              +-->N95-1411 B2 : Pancreas, block 2
                          |
                          +-->N95-1441 B2 1 : Pancreas, block 2, slide 1
                          |
                          +-->N95-1441 B2 2 : Pancreas, block 2, slide 2
```

Once specimens have been mounted on glass, prepared slide labels contain:

Cassette # / Accession #
Block #
Mayo Clinic #
Date

"Surgical Pathology Assistants" prepare the slide labels, instructed by the dictating surgical pathologist.

If a referral is done on a slide, the turn around time in usually one day. When a referral is sent out, the "send-out" portion of the system keeps track of it. In addition, it prints out a letter to go with the slide, and has the capability to mark "return" if the slide is to be returned. The number of slides lost or broken is less than one percent, Ms. Madjidi said.

The Histology module, for any given case, could show how many and what kind of tissue specimens were logged into the system, what stains were ordered for those specimens, could print cover reports for referral slides, and could print "outstanding slides" reports for missing/referred slides.

CoPath can perform searches on the following fields: specimen number, medical record number, patient name, attending physician, and time frame.

CoPath handles CPT (Current Procedure Terminology) codes which are necessary for billing. The billing interface to the HBO Clinstar billing module was via manually transported diskette rather than a direct interface. CoPath also automatically assigns SNOMED codes (topographically based) by reading[i] the logged diagnosis. Dr. Madjidi explained that SNOMED codes are required by their "Tumor Registry" located at SMO. The registry compiles statistics which are forwarded to national registries. Pattie did not elaborate on the kinds of statistics generated or for what purpose.

When conducting a search in the system, specimen number, attending physician, medical record number, or patient's name can be used to pull up the information you are looking for. In addition to this, SNOMED coding can be used. SNOMED codes define a specimen by diagnosis and topographical location. For example, and SNOMED code will tell you that a tissue is from the breast and that it is carcinoma. SNOMED has thousands upon thousands of codes for all of the possible information necessary. Ms. Madjidi said that this was good for tracking specimens to search for later on, and that it also keeps track of things internationally. SNOMED is intended for use by the Tumor Registry Department, which must report to government and state agencies. All hospitals have their own Tumor Registry Department which must report to a main hub. Therefore, everything is reported. Ms. Madjidi also added that SNOMED searches are very quick.

As far as billing is concerned, fee codes are entered when the specimen comes in. This is the only link to the hospital information system they have. Standardized codes are used for insurance company reasons.

Records are not archived, Ms. Madjidi said. Everything since 1991 has been kept on a database, and there is still plenty of room left. Everyday back-ups are performed also. This is done on site, but at a different location. As far as accessing records from three or four years back versus a month or two ago, there is no difference in search and retrieval time. It takes a couple of seconds at most to pull one up.

The only wish-list items that Dr. Madjidi has for CoPath is, 1) a light-pen, of which she has used before on other systems, but she is not exactly sure how it might be integrated with the CoPath user interface, and 2) a windows implementation. When asked for CoPath weak points, Dr. Madjidi could think of none.

The HBO&C system was briefly cited. It is known that at least the HBO Clinstar (a "laboratory system", said Pattie), and Trendstar are installed at SMN. From KSC in-house surveillance literature[ii], the TRENDSTAR is a "decision support system that guides decision making. It offers `point and click' access to diverse sources of information, powerful analytical functions and extensive reporting and presentation tools. TRENDSTAR complements HBOC's

STAR and HealthQuest financial and clinical products by providing managed care contracting and monitoring, quality and case management, budgeting, forecasting and strategic planning solutions to support enterprise-wide decision-making."

Application Software:

- o Hospital Systems Library
- o Clinical Cost Accounting
- o Contract Payment Advisor
- o Management Cost Accounting
- o Marketing Systems Library
- o Resource Utilization Analyst
- o EpiTREND Reporting System
- o TRENDPATH

"HBO Clinstar" was not in any of our literature. If it is a "laboratory system", then likely it is the STAR Laboratory, described as, "an advanced system designed to ensure quality of outcome through far-reaching communication, data integrity and management control. Extensive quality control features help ensure the reliability of test results and the appropriateness of care delivered."

Application Software:

- o Order Management
- o Test Processing
- o Patient Inquiry
- o Patient Result Reports
- o Quality Control
- o Administrative/Management Reports
- o Surgical Pathology
- o Advanced Microbiology
- o Advanced Blood Bank
- o Contract Management

One last final note, Ms. Madjidi said that the hospital produces about 15,000 Pap Smears and 10,000 surgical slides annually. From September 1st through the 15th, she was able to pull up on the computer that 769 surgical slides had been produced.

2.6.2 Pathology

After quickly reminding Dr. Jolma, Medical Director of Scottsdale Memorial, of the nature of our visit, he kindly agreed to spend 30 minutes with us due to his busy work schedule (however, due to his interest later, we stayed a full hour). Our focus was to present our vision of the ideal telepathology system and allow Dr. Jolma to critique it. Further, we gained insight into his work and asked him to explain to us what kind of automation could improve his work. He was helpful in answering our questions and gave us some good ideas to keep in mind.

When asked what areas of technology he would like to see improved, Dr. Jolma stated that it would be nice to be able to pull up prior images from a past biopsy on the screen to compare it to a current slide in the office under the microscope. He said that this would be especially helpful with bone marrow and leukemia. If a patient was receiving chemotherapy for leukemia, he could pull up an image from the past and compare it to what he has in his office to see the progress that has been made. He also felt that the whole area of cytology could benefit from this too e.g. (with abnormal Pap smears). Other areas where history and cross-correlation of data would be helpful include; Colitis, Liver, Kidney, biopsies, and D&C's. He stated: "It would be interesting to see how D&C's respond to hormone therapy."

Dr. Jolma agreed that a market exists in rural hospitals such as one he knew in Paysen, AZ. He cautioned us not to indicate himself as a reference in case we followed up in Paysen.

Dr. Jolma next discussed what he felt would be "helpful computer aids."

The first aid would be to display image-based biopsy history of a patient on screen side-by-side. Currently he must swap several slides in and out from under the microscope, while retaining visual/heuristic knowledge of removed slides and comparing these to the current slide under microscope. An example was bone-marrow biopsies from patients undergoing chemotherapy--would like to quantify differences in specimens over time, to determine the next treatment step. Placing the imagery side by side simulatenously on a computer screen with sufficient resolution would be very helpful.

Dr. Jolma felt that the whole area of cytology could benefit from such computer-based technology. Had mentioned something about CLEA requirements but we didn't have time to determine what CLEA and its requirements are.

Dr. Jolma also stated that being able to pull up a history of slides on the screen at one time could be helpful because there tends to be observer variation among different pathologists as to what the degree of abnormality is.

Regarding voice recognition, Dr. Jolma felt that there was a place for it in the future. He had observed an old Kerswell demonstration in Boston where there was still much work to be done. What he has experienced in the past has been a struggle to use, but if advancements have been made to the point where you don't have to talk robotically and change your vocation, then he thinks voice recognition would be terrific. First of all, it would reduce some costs and reduce turn-around time with the clerical help. They are to dependent on the clerical help. This also holds up the diagnosis and the release of the patient from the hospital. Voice, he said, would speed things up and reduce the cost of secretaries. Auto-dictate to script would be great. He does not feel there would be any resistance by pathologists to use it; they are just waiting for a good system to come around. We asked him if he had any comment on the physical interface used in voice recognition systems. He indicated that a head-set or microphone was no problem at all for him to use as pathologists must get used to microphones on dictation machines and other physical interfaces such as the microscope. Thus, as long as the using a given physical interface enhances work without degrading concentration, such an interface is desired and accepted. Dr. Jolma showed preference for a voice activated system since not too much commotion goes on around them to disturb it.

Dr. Jolma said that he desires more collaboration among local pathologists for diagnostic concensus building on hard-to-diagnosis specimens. He explained that currently, each Pathologist has his own office and works there with his, "pile of glass". Often he would like to take a difficult specimen to a college and get his opinion. We pointed out that such unannounced drop-in visits with other Pathologists are probably not always at a convient time; it would be much better if the consultation could be done digitally through a local email system so that the consultant could review the digital specimen at his earliest convienience, much like voice mail and email messaging works to increase efficiency. Dr. Jolma raised no objections. Not only would this eliminate a Pathologist's footwork between several other pathologist's offices, but would also furnish rapid diagnostic turn-around since several pathologists could examine the specimen simultaneously and forward their results to Dr. Jolma's workstation. Dr. Jolma said that such technology will become increasingly attractive since, "the lawyers have discovered the Pathologist," and increasing 'patient vs. the pathologist' legal activity is, "driving this technology."

Dr. Jolma would also like to see pathologists break away from the office and work more together in a big room. This would facilitate communication as well as consulting, and

pathologists would be able to trade slides around. They would be working much closer together than alone, as they are in the present office situation.

Dr. Jolma feels that the true savings from telepathology will occur with the small, remote hospitals which may not employ a pathologist, but only a technician. Instead of a pathologist having to travel out to the hospital once a week to diagnose slides, the image could be sent to them over the telepathology lines. When queried about the desired resolution for a guide image, Dr. Jolma suggested that of a red blood cell (6 to 7 micron).

As far as using telepathology for referrals, Dr. Jolma prefers to have the slide in his hand. If an image were to be used, the sending pathologist may limit the areas to be viewed, causing a biased diagnosis. Dr. Jolma strongly felt that he would like to see the entire slide at high-magnification, and not just a small ROI section. In fact, he would hesitate to sign his name to any ROI another pathologist had subjectively produced, forwarded to him, and required his proffesional diagnosis. Somehow, the entire slide should be online for Dr. Jolma to review--either the entire slide must be digitized or be remotely available in a physical sense for real-time scanning and analysis. This necessity may suggest that Kensal seek to design a mechanical carousel/cabinent of some kind that could robotically load/unload slides, requested from remote sites, into the PC microscope. Maybe a heirarchy of robot linkable slide containers of varying sizes could be designed, with each suited for various client requirements. Alternatively, a digital container, made solely of RAID[iii]-technology, with the capacity to store a score or so of slides who have been completely digitized with the LSDA and the lensed microscope for temporary on-line storage could also be produced--however this would require a tremendous amount of high-speed automated rectilinear scanning at all levels of magnification over the entire slide. Further, "stiching" software[iv] would need to be developed to remove lens distortion and piece the digital pictures together into a distortion free resulting image.

With the slide in his hand, there is no selective choosing of areas to look at by the other doctor. Medical/Legal restrictions will be a driving force. If we can down-load prior images and show areas observed - this may alleviate some of the concerns of the pathologists.

Interestingly, Dr. Jolma had a multi-viewer, pipe-line-esq, microscope on his desk-top for at least three simultaneous viewers. We did not ask him how he used this--perhaps one use is for diagnostic consensus building? If so, he must coordinate schedules for one or two other pathologists to hold a group review session. The advantage to such a session is that all can express their opinion verbally, immediately, and together while seeing the same image, with a session moderator--Dr. Jolma (he controls the group's visual ROI). However, the scenario may be likened to the different collaborative results yeilded when comparing the standard "brain storming" session with a "nominal group technique"[v]--the latter yeilds greater productivity and more results in less time. The nominal group technique is analogous to our technology used in a non-real-time fashion[vi], namely emailing image-based querries to the queues of other Pathologists for review and later merging the results-data at the orgin of dissemination for concensus overview. The cycle could be itterated to reach for unanamous agreement. If after several itterations, unanimity is not reached, then the specimen could be reffered to a outside specialist of that specimen's organ system.

Dr. Jolma's suggestions were that we learn more about Image Analysis, and them "marry" it to telepathology. Dr. Jolma strongly suggested that we consider including image-analysis with our workstation. "If you can marry image analysis with telepathology, I think this would very good." Dr. Jolma next spoke of image analysis of nucliea and suggested that we check MEDLINE for current papers on Medical Image Analysis. There have been some NIH studies in this area. Image analysis involves looking more at the nuclear material, such as for cancer, and has inroads in flow cytology (which does better than the human eye). Nuclear Image Analysis involves the

computer recognizing the characteristics of the nucleus and if it is Bi, Tri, or Tetra-ployed. This could be a new area of interest for Kensal, marrying telepathology and Nuclear Image Analysis.

2.6.3 Conclusion

Dr. Jolma yielded useful input in a timely and forthright manner; this input spawned some very interesting possibilities, some previously conceptualized (revisitation should cause us to think again more seriously about these possibilities). In review, some key points I feel need further discussion are as follows:

Follow up on nuclear image analysis with MEDLINE; if our technology can be meshed with nuclear image analysis, get back in touch with Dr. Jolma for more specifics.

Discuss requirements of scanning entire slide at all levels of magnification--discuss the required two phase stitcher; phase one to remove lens distortion and add color balance, etc.; phase two to convert a slide coordinate into an ROI image through stitching together several phase-one corrected capture images. NOTE: it may be better **not** to create a single large image from scanned pieces but rather to stitch pieces together "on the fly" for immediate presentation, as is evident in Preston's Infinite Roam technology and Adobe Photoshop's approach to updating large images on screen.

Discuss robotic sample containers for on-line remote selection and loading into PC microscope.

Discuss potential for an "in-house" LAN based telepathology system for diagnostic consensus building between offices of pathologists through the nominal group technique.

2.7 UMC/Cortex Medical Management Systems demo by Judith Krebs and Mark Stevens (cyto geneticist)

Mrs. Krebs was coming into Tucson to check on the newly installed Gold Standard System at University Medical Center. She agreed to meet with us and perform a demonstration so we could see how their system worked. Cortex Medical Management Systems is very interested in linking up with us so that they may include imaging into their system.

On 9/18/95 Arvie Lake, President of Boeckeler Instruments, Inc., Kendall Preston and the writers visited the University Medical Center (UMC) for a Cortex *Gold Standard* demonstration arranged by Laurie.

The system uses a Novell and Ethernet networks and gives reasonably fast response. Judith has found that doctors are definitely becoming more interested in computers. There is a trend for doctors to dial in from home, sign out reports with their electronic signature capability, etc. In addition, there system has 3 to 4 levels of security built in including a login code, a Cortex identification number, a password for pathologists who type up the reports, and then a forth level code for someone who wants to change a report.

Approximately 5 universities, 6 to 10 big clinical labs and 12 to 18 mid-size hospitals are using their system. Numerous smaller reference labs, cytology labs near hospitals are going to the *Gold Standard.* There are about 75 users of the system which handles Cytology, Anatomical, and Surgical Pathology. Most of these institutions use it for anatomical/surgical pathology purposes.

When asked about the acceptance rate of using computers, Mrs. Krebs stated that there was a greater acceptance among younger doctors than older ones. The older doctors tend to like to stick

with what they are most familiar with, which is having slides and reports in their hands. She estimated that out of their 75 installations, about 50% actually use the computer system.

The system provides several levels of security depending upon the application, For example only a pathologist can certify changes in his report. There are two major files, the Case file and the Case History file. The Case History file is on-line, (they have about 5 years of storage on a one Giga-byte hard drive with only 18 months of records). They encourage everyone to transfer information into the case history file after eighteen months. They do not archive optically yet, records are backed up on DAT tape using Colorado Mountain Software.

Cortex recommends backup; daily, weekly, monthly, and every half year. The system uses a report writer "Advanced Revelation". There are several ad hoc reports that are standard or custom reports can be easily generated. Note copies are in Lane's file. The system we viewed was their DOS system. They are in the process of switching over to a Windows system which incorporates the Windows 95 system or the 3.1 system. Mrs. Krebs said that she expects about one-third of their users to convert to the Windows system. They have two beta sites happening right now (chosen because of an interest by the beta sites), and hope to have a fully functional Windows system by Christmas. UMC is using all modules of *Gold Standard* which include Cytology, Histology, Autopsy, Billing, Support, and Query.

The system holds all prior patient history (data entered by Cortex). Records can be accessed by Name, Medical Record, Birth Date, SSN. and Admission number. They can toggle over to LIS etc. through a custom interface. The Pathology system uses status codes (1-9) to help simplify data taking: 1-case entered, 2-history printed, 3 gross entered, 4 gross printed, 5 diagnosis entered, 6 Diagnosis printed, 7 case signed but not locked down, 8 case signed and locked down, 9-billed history. The system doesn't store the slide number but only the case number. All information is on line in "gross" format but printouts use various codes. CPT (Current Procedure Terminology), custom billing codes, ICD for diagnosis, and SNOMED are all used. As far as case numbering goes, the system itself can assign a case number or a person can type one in. The system also contains SNOMED codes for tracking, but Mrs. Krebs stated that she thought SNOMED was dying out because it tends to be very time consuming to work with. There are only four basic reports, but a facility may have several versions of each. there are also Work in Progress, Case History, Regulatory Report, Statistics Reports mostly in Cytology. Most reports are one page in length, however the Autopsy report may be 5 to 15 pages in length. UMC has written its own custom 4-page cytology report. An HP LaserJet printer is used for output. "Orbit Enterprises" provides their digital signature software which works well. The new "Windows" software will allow then to more easily Fax reports.

Cortex is working on speech recognition but today it is still a problem for pathologists. Third parties are working to allow interface with images. The company has 50 years in pathology and started its first sales in 1987. The current DOS system is very stable. Pathologists can learn the system very quickly and the Transcriptionists do the rest. Searches are very fast due to their special status codes and indexed fields. They have a relational database with variable length fields. It takes about 10 seconds to call up a case. Cortex is unique in that its staff is small but very available. Training is a big part of their effort. They are not limited by size, most systems have several workstations, but an Australian installation has 50 workstations. They also have active well trained users' committees and a users' group who help decide on new features. one third of their sites were represented at a recent users conference. They are planning to be on the Internet shortly. There are a dozen competitors of which only 2 or 3 are serious. About 4 out of 10 or 12 pathologists are fully trained after the first training session while another 5 need additional help. The system employs several beneficial features. There is an extensive help list. A keyboard template shows the application of each of the function keys which can be significant time savers. The overall HIS uses a Sunquist system and has a LIS module. An "AO" module interfaces between the Sunquist system and the old Anatomical Pathology Module which is written in

Mumps. Several of Cortex's customers use the Gold Standard instead of the HIS pathology module supplied by the HIS vender. "Since Cortex specializes-they do it well".

The Cortex system has four one-way interfaces, One to IDX for billing, two to ADT for admissions, three to the HIS, four to Sunquist for billing. Cortex can also receive information from Sunquest. Visual Basic is being used for the upcoming "Windows" version. The Advanced Revelation uses R Basic and R&R Report Writer and Crystal Reporter. HL7 is used as the standard interface. Bar-coding is used for cassettes and patient information and Requisitions. Scantron is used to enter Cytology reports.

UMC runs 2000^+ pathology reports a month plus cytology reports. A typical day was counted with a total of 166 reports. There are about 10 to 30 cases per pathologist per day. The word processor can also spell check. MS Word will be used with "Windows" so new words can be added easily. The current system uses electronic billing to reduce paperwork.

2.7.1 Conclusion

Cortex seems to have made good inroads into pathology information systems and will be a company to closely watch if they can be successful with "Windows 95". Perhaps we can help them with imaging and interface to the PC Microscope Pathologists' workstation.

2.7.2 Attachments

In Lane's File Cabinet under UMC:

1. UMC Surgical Pathology Report, 3 PPS
2. UMC Cytology Report, 1 PPS

2.8 Mayo Clinic, Scottsdale, AZ

One of the three "blind" slides received from Mayo Clinic (from one of Dr. Weiland's associates) was scanned in-house and printed on the color "Fiery Cannon" at 200 dpi (11" x 17") format. The LCD scan produced 100 vertical lines per inch with a maximum of 1024 pixels per line. Three scans were conducted using red, blue, and green filters with merging and matching accomplished in "Adobe Photoshop".

Some color mis-registration was observed (cyan). This did not significantly hinder the reading of the "Guide Image" from the print. With observation the vertical scan lines (100 per inch) could easily be observed, the 200 lines per inch (horizontal lines) are observable to a nearsighted person especially without glasses.

Upon the opening of the print, Dr. Weiland did a quick scan of the guide image and observed three samples of tissue. The center sample had already been marked (blue ring of dots) by a previous observer. Within one minute, he stated that "He knows what he wants to see." In another half minute he stated: "He has everything that he wants to see for diagnosis." It then took one minute to carefully mark the print with a sharp X for each Region Of Interest and the desired magnification level.

There were seven ROI's chosen all at an additional 20x. Two of the seven sites were also requested at 40x for a total of 9 magnified images. Four of the sites were in the previously marked image, two were on a second sample of tissue while the last was on a third sample.

Dr. Weiland stated that he thinks he knows what the guide image shows, but reserves the definition until he sees the final magnified images. He had no complaints as to image quality of the print.

2.8.1 Conclusion

It appears that this approach will be satisfactory in simulating Telepathology with our present equipment. Although more time and cost is required for the prints, our general approach is verified. Having the prints as documentation is a significant advantage. When we stated that the final PC Microscope will document all portions of the guide image that the Pathologist has observed; Dr. Weiland was enthusiastic in his interest and support.

2.9 References

3. RAPID PROTOTYPING (by Steven R. Lange, Boeckeler Instruments, Inc.)

The Boeckeler TSS1 (Telemedicine Support System for Pathology) consists of workstations working together over an ISDN telephone link. The Boeckeler TSS1 can work in a single (stand alone) workstation, or in a dual mode configuration. All workstations can send and receive images. This allows one pathologist located at one part of the country to send or receive images from remote locations. All of the workstations use state-of-the-art, off the shelf components.

The workstations have the following features:

- Windows touchscreen control of X,Y,Z stage and all functions
- Non-complex screens
- Multitasking computer system
- Graphical User Interface screens and controls

Two telepathology instruments were fabricated for the purpose of conducting experiments to:

- Test the productivity of a two telepathology instruments for the purpose of conducting experiments to compare ordinary microscopy in conducting an examination.
- Test the performance of the pathologist using the instrument in its telepathology mode for conducting the same or similar examinations.

Each workstation consists of the following:

- Microscope including stand, objectives, eyepieces, illumination and condenser with provision for computer control of focus, objective power, condenser and illumination.
- Stage with motorized X, Y motion with manual/computer control.
- Full slide scanner including Kendall Preston Jr's patented "lensless microscopy" technique to obtain a "guide image".
- Personal computer with touchscreen activated graphical user interface and image capture/display, compression, dcompression, storage, retrieval, transmission and receipt capabilities for images and verbal dialog.

3.1 Function

The primary functions of workstations:

- Obtain guide image by scanning the entire microscope pathology slide and capture and display the images in high-resolution color.

- Scan user-designated "areas of interest" at user-selected high magnification from 2X to Acquire user-designated high magnification images of sample under study.
- Utilize JPEG software compression algorithms to compress/decompress images for storage and/or transmission.
- Decompress and display stored/transmitted images.
- Provide for user entry of six-digit surgical number for each image.
- Transfer images, location, coordinate data, magnification and other pertinent subject data to the second work station over ISDN high-speed telephone lines.
- Allows for simple, intuitive user control of all system functions utilizing a touchscreen control interface.
- Record the X,Y location of the stage and the magnification once per second while the system is in the single-station mode and record to a file for the purpose of analyzing the motions of the pathologist.
- Record and playback of verbal dialog attached to image files.

3.2 Operator Functions

The operator/user manually loads a pathology slide into the carrier and presses the appropriate button on the touchscreen to initiate the scanning of the slide. The user can change the position of the X, Y stage, and also fine focus the microscope image on the CCD camera with computer assistance.

The operator/user must manually turn the system on and off when required, and manually load or remove a slide from the carrier when requested. The operator will visually check and make sure the slide is loaded in the carrier. The operator/user will follow the instructions from the TSS1 program and may get help from the on-line help menu. The computer will boot up in the Boeckeler TSS1 Main window. If the user wants to exit the Boeckeler TSS1 program the user can do so by pressing the *Exit to Windows NT* button. The Boeckeler TSS1 is not fully automated and the operator will be asked to load slides into the carrier and make minor adjustments (focus).

3.3 Organization

The program is developed around screens. Each screen allows the user to only accomplish a few selected tasks. Further, the program as a hole is organized such that a user is directed down a path where only certain options are available to him. The options are selected in the Main TSS1 screen (the first screen). Each path selected in the main menu is designed to accomplish a task in either the single (local) station or multiple station modes of use. In the single station mode, the operator is interested in examining slides and making a diagnosis as quickly as possible while examining enough of the slide to be complete. In the multiple station mode, the operator is interfacing with another operator at another station operating in a consultant mode. In this case, each operator must respond to information requests coming in both directions. The requests take the form of files that are sent from one station to another using the ISDN phone lines. The requests appear the form of an electronic mail box where certain files will insert a message on the main menu (e.g., “Images Requested”). The operator is expected to see the request and respond by pressing the appropriate button on the main menu. After pressing the button, the user will be led down a path, screen by screen to fulfill the request. The use of the stations can be best seen graphically as in Figure 1 & 2. Figure 1 illustrates the single station mode where the examiner is just viewing the slides to make a diagnosis. Figure 2 illustrates a mode where a requesting station sends images to a consultant for examination.

The flow shown in Figure 2 is simplified into its most direct form. In some cases, the requesting station may send both the guide image and several high-mag images that the operator thinks are needed for a diagnosis. The consulting station may see everything he needs in the images that are sent and he can then send a diagnosis back to the requesting station eliminating

many steps. Another scenario may entail several requests for high-magnification images that are needed for a complete diagnosis.

The final path involves only viewing old stored images and playing the sound files accompanying these images. A file server that shows the thumbnail images for each stored image is accessible using the *Single station recall old images* button on the main menu. After the thumbnail is viewed, the operator can view either the guide or accompanying high-magnification images in full size.

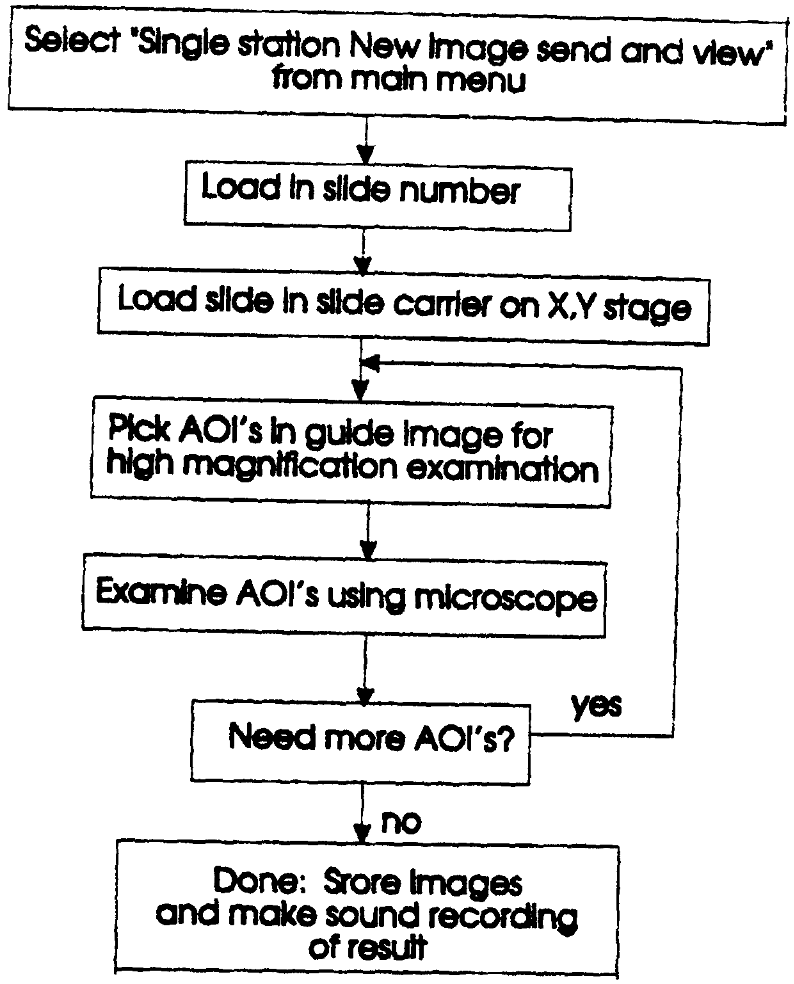


Fig. 1 - Single station (local) operation.

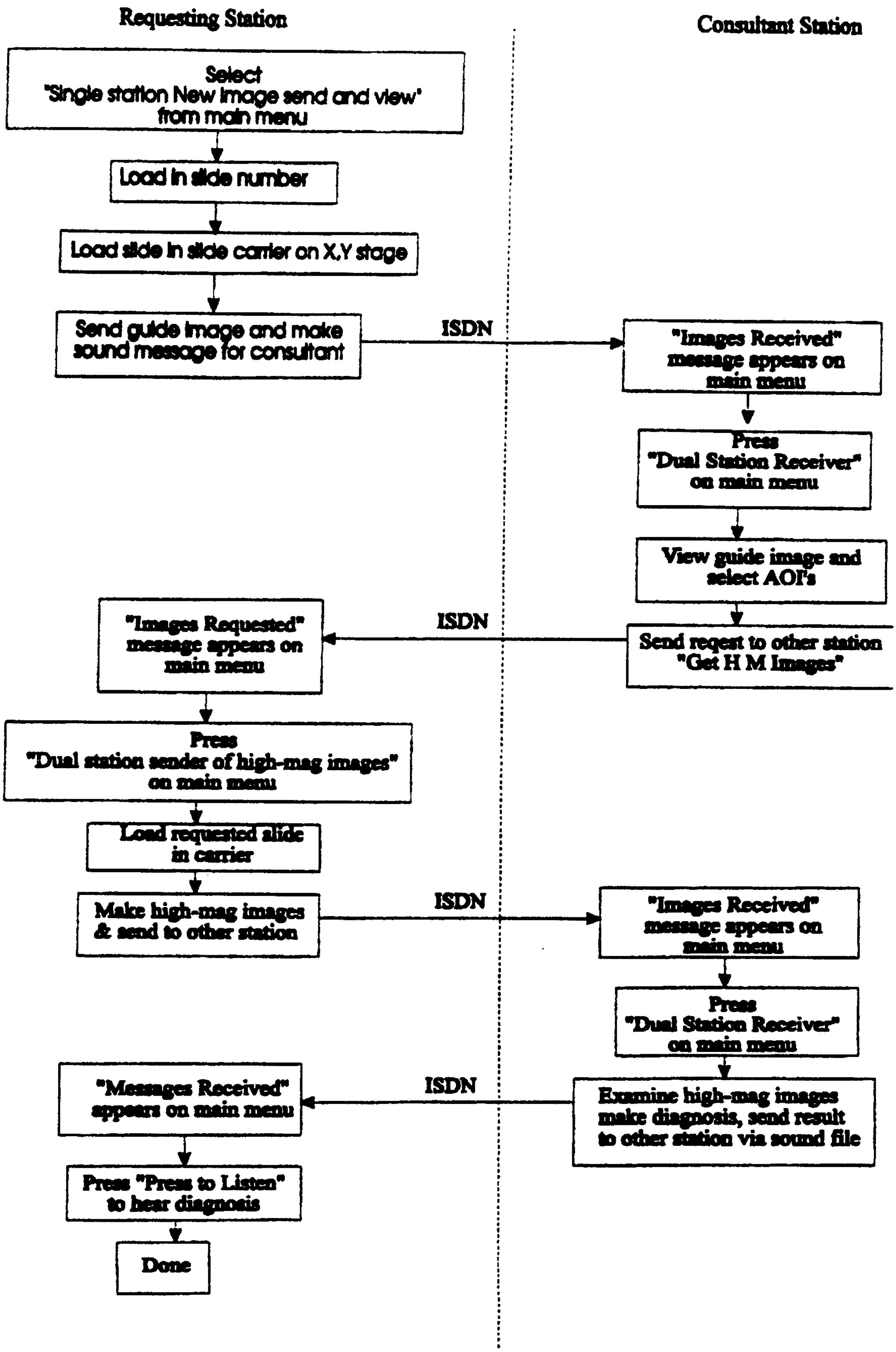


Fig. 2 - Multiple station operation.

3.4 Screen Operation Description

This section describes the operation of each of the screens and what happens with each of the buttons.

3.4.1 "Main TSS1 Window" Window

Synopsis:

This main menu controls the direction of the program. Pressing any one of the four main buttons puts the operator down a path to accomplish a task. The description of the paths are explained below under the button for each path. Messages will be displayed here when information from other stations have been received inviting the operator to act on those messages.

Operation:

"Single Station Recall Old Images"
This button will bring up a screen which will allow you to review stored guide and high-magnification images.

"Single Station New Image Send and View"
I) Examination: This button will let you scan a new slide to create a guide image. You can use that guide image to locate Areas Of Interest "AOI" and then direct the system to find those AOI's on the slide with the microscope. Thus, you can do a complete examination of the slide here.
II) Consultation: You can also prepare information to send to another station for consultation. This information will consist of the guide image, and may also contain high-magnification images and sound messages to accompany each of the image types. You can direct any or all of this information to be sent to another station.

Once received at the other station,. a message "Images Received" will appear on their screen indicating that a consultation request has been received. If just a guide image was sent to another station for consultation, it is expected that the consultant will request one or more high-magnification images to be sent to them to aid in the diagnosis. After they receive the guide image and view it, they will select a number of AOI's and request you to make the requested high-magnification images. Their request, when it comes to your station, will have the program display the message "Images Requested" on the main menu. Your job then would be to use the "Dual Station Sender of High-Magnification Images" button and follow the directions indicated in the windows that follow. Once you have made the requested high-magnification images and sent them to the consulting station, that station will again display the message "Images Received". The consultant will press the "Dual Station Receiver" button to view the new high-magnification images. If they complete the diagnosis, they cab send a message back to you indicating the result of the examination. Your station will display the "Messages Received" sign with the "Press to Listen" button activated. You can press the button to listen to the result of the consultation.

"Press to Listen"
A sound message has been received from another station. The sign "Messages Received" will appear on the main menu with the button "*Press to Listen*" activated. The message should contain the diagnosis from a guide and set of high-magnification images sent for consultation.

"Dual Station Receiver"
This button will be activated when your station is requested to be a consultant on slide images sent to it by another station. The message "Images Received" will be displayed below this

button. If you press this button, you will be guided through a process to examine the images sent to you from the requesting station. This process may include requesting more high-magnification images from AOI's that you have selected along with sound messages helping the requester obtain the needed information. The requester may have sent you a sound message accompanying either guide or high-magnification images. You may have enough images to make a diagnosis. in which case, you can send the result back to the requesting station via a sound message.

"Dual Station Sender of High-Magnification Images"

This button will be activated when you have been requested to make high-magnification image to aid in a consultation you have requested from another station. By pressing this button, you will be directed to load in a specified slide number and follow directions from the requesting station. The information from the other station will direct the microscope to examine AOI's where you will be expected to create and send the needed high-magnification images to the other station. You may receive a sound message to aid you in identifying what the consultant is wishing to examine.

"Exit to Windows"

This take you out of the program and into Windows NT. A warning message will appear to give you the choice to return here.

Usage:

Select an operation based upon your desires or the messages received from other stations. The program will guide you through the steps to accomplish the task you started here.

3.4.2 "Surgical Slide Entry Number" window

Synopsis:

This slide ID number gets tagged to all data allocated with this slide, including the guide image, all high-magnification images, message files, request files, etc. In addition, the ID-number will appear on each subsequent screen associated with this slide.

The files created from this slide will be stored on hard disk under a directory named the same as the slide ID number (e.g., C:\314567GID\314567.GID). High-magnification images will be stored under a sub directory (e.g., C:\314567GID\314567-01HIM\314567-01.HIM). Using the Windows file manager in NT, you can see the directory structure for each slide. This structure makes it easy to copy the entire data for each slide to a backup device such as the Syquest 270 Mb cartridge drive.

Operation:

"CLEAR"
If you need to change what you input.

"RETURN"
Will take you back to the main menu.

Usage:

You need to type in a six-digit slide identification number and press "OK".

3.4.3 "Load Slide in Carrier" window.

Synopsis:

You need to load the slide into the slide mount and press "OK". The slide should be oriented as shown below, with the information tag farthest away from you.

Make sure that the slide is securely positioned against the sides of the slide mount and parallel to its sides.

Operation:

"Return"
Will take you back to the previous screen "Surgical Slide Number Entry" window.

"OK"
Starts the scan of the guide image.

Caution:
The system, at this time, will not do a test to see of the slide is properly mounted in the carrier or even a slide is mounted in the carrier.

NOTE: Damage to the scanner could occur if the slide is not mounted properly.

NOTE: The slide must be clean and dry for a good guide image to be scanned. Make sure all debris is removed from the slide surface.

3.4.4 "Guide Image Display - Select for High Mag" Window

Synopsis:

This window displays the guide image that just scanned for examination. This means that you should look at the image and select which areas of interest "AOI" you need high-magnification images to make a diagnosis. The AOI you select will be written into a file and when you leave this window will direct the microscope to find for you to make a diagnosis.

Operation:

"Record Message"
Will record a recorded sound message to be stored with this guide image to describe what information is associated with it.

"Quit"
Will take you back to the main menu. A dialog box will appear informing you that this will take you back to the main menu and you will lose all the AOI's you have identified so far.

"Get HM Image"
Will have the system move the slide to the first AOI requested under the microscope. The requested information includes the location on the guide image and the magnification at that location. Up to 99 AOI's can be identified for each guide image. If no requests have been made and this key pressed, a dialog box will appear notifying you that no locations have been selected and bring you back into the window again.

"Record Image"
Stores the guide image on hard disk.

"Xmit G Image"
Stores the guide image on hard disk and transmits it to the other stations for examination. You should also record a sound message to send along with the image to indicate what the receiver is to look for in the image and to relay other important information concerning the image.

"Select Mags"
Brings up a display box that allows you to select the magnification for an AOI. You can select either a 2X, 4X, 10X, 20X, or 40X for each location. The size of the buttons for each magnification represent the area on the guide image that will be examined with the microscope.

"Guide Image"
The display of the guide image is only a small part of the full slide. To move the image around to see more of it, you can press on the guide image and the image will move in the direction you press from the center of the image. A smaller image (thumbnail) of the whole guide image is displayed in the upper right part of the window. The area seen in the large display is identified in the thumbnail with a colored box.

Usage:
Pick AOI's by moving around the display of the guide image until a region is seen where a high-magnification image is required. Press on the "*Select Mags*" and position the appropriate magnification over the AOI. Repeat until all AOI's are identified. Then press "Get HM Image" to start the examination with the microscope of the AOI's.

3.4.5 "High Magnification Image Window" Window

Synopsis:

This window displays a real-time high-magnification image for examination. You can move the slide around, change microscope objectives, save, and send the image. You can make a diagnosis from this slide, if desired, and make a sound file indicating the diagnosis, or send the image and sound file to another station for consultation.

Operation:

"*Record Message*"
Will record a recorded sound message to be stored with this high-magnification image to describe what information is associated with it. This message will be sent to another station along with the image if "*Send Image*" is pressed.

"Quit"
Will take you back to the main menu.

"Next Location"
This will move the microscope to the next high-magnification location you previously requested (Loc. will increment by one up to the # of Loc's as displayed in the window below the image.) If this is the last location, the program will take you back to the "Guide Image Display, Select for High-Mag" menu to select other AOI's if desired.

"Select New High-Mag Images"
This will take you back to the "Guide Image Display, Select for High-Mag" menu to select other AOI's if desired.

"Record Image"
Stores the high-magnification image on hard disk. A dialog box will ask you if you want to record a sound message to accompany this image.

"Send Image"
Stores the high-magnification image on hard disk and transmits it to the another station for examination. A dialog box will ask you if you want to record a sound message to accompany this image. The sound message sent along with the image should indicate what the receiver is to look for in the image and to relay other important information concerning the image.

"2X. 4X, 10X, 20X, 40X"
These button allow you to select the magnification currently being used with the microscopes.

"High Magnification Image"
The display of the high-magnification image is only a small part of the full slide. To move the image around, you can press on the high-magnification image and the slide will move so that the location you press will become centered in the display. A smaller image (thumbnail) of the whole guide image is displayed in the upper right part of the window. The area seen in the large display is identified in the thumbnail with a colored box.

"Focus"
This slider will allow you to focus the microscope. The speed of movement is scaled to the magnification used.

Usage:
You can move the slide around, change microscope objectives, save, and send the image. You can make a diagnosis from this slide, if desired, and make a sound file indicating the diagnosis or send the image and sound file to another station for consultation.

3.4.6 "Request to Load Slide" Window

Synopses:

You need to load the slide with the slide number requested into the slide mount and press "***O K***". The slide should be oriented as shown below, with the information tag farthest away from you.

Make sure that the slide is securely positioned against the sides of the slide mount and parallel to its sides. You will be directed to make the number of high-magnification images shown in the window for this guide image. After you load the slide and press "***O K***", the system will find the first location on the slide requested and position the slide under the microscope at that location and with the requested magnification. After you have made any necessary adjustments to position, focus, and magnification, you will press "***Send Image***". The system will then move to the next requested position and repeat the operation until all the requested positions have been sent.

If more than one set of high-magnification images have been requested for other guide images, the number of guide images requested will be output in the window. The above sequence will have to be performed for each of the groups of high-mag's requested.

Operation:

"Ok"
Press when the slide is loaded in the carrier. The system will move the slide to the first AOI identified for examination.

"Return"
Will take you back to the main window.

"Play Message"
Will play any message that was sent along with the request to help you in making the requested high-magnification images.

Usage:
The system, at this time, will not do a test to see of the slide is properly mounted in the carrier or even a slide is mounted in the carrier.

> **NOTE:** Damage to the scanner could occur if the slide is not mounted properly.
>
> **NOTE:** The slide must be clean and dry for a good guide image to be scanned. Make sure all debris is removed from the slide surface.

3.4.7 "Received Images" Window

Synopsis:
Allows the operator to view received guide and high-magnification images.

Operation:

"View Guide Image"
Takes the selected guide image and displays it in Screen 8.

"View High-Magnification Image"
Takes the selected high-magnification image and displays it in Screen 9.

"Quit"
Takes you back to the main menu.

Usage:
When the screen is called up, a list of all the received guide images will be displayed on the left window "Guide Images". If one is selected by pointing to the file name, a thumbnail of that guide is displayed in the "Guide Image" window. Also, any high-magnification image files that were stored that accompany that guide image will be listed in the right window "High Mag Images". If one of the high-magnification files is selected by pointing to the file name, a thumbnail of that high-magnification is displayed in the "High-Mag Image" window. Either the guide or high-magnification image can be displayed in a full screen by pressing either "View Guide Image" or "View High-Mag Image".

3.4.8 "Guide Image Display - Select for High Mag" Window

Synopsis:

This window displays the guide image that was send from another station for examination. This means that you should look at the image and select which areas of interest "AOI" you need high-magnification images to make a diagnosis. The AOI you request will be written into a file and sent to the other station to have the high-magnification images made and sent back to you.

Operation:

"Play Message"
Will play a recorded sound message from the other station, if one was recorded. This information may help you in making the diagnosis and picking AOI's.

"Quit"
Will take you back to the main menu. A dialog box will appear informing you that this will take you back to the main menu and you will lose all the AOI's you have identified so far.

"Get HM Image"
Will send the requests for AOI's to the other station. The requested information includes the location on the guide image and the magnification at that location. Up to 99 AOI's can be identified for each guide image. If no requests have been made and this key pressed, a dialog box will appear notifying you that no locations have been selected and bring you back into the window again.

"Select Mags"
Brings up a display box that allows you to select the magnification for an AOI. You can select either a 2X, 4X, 10X, 20X, or 40X for each location. The size of the buttons for each magnification represent the area on the guide image that will be examined with the microscope.

"Guide Image"
The display of the guide image is only a small part of the full slide. To move the image around to see more of it, you can press on the guide image and the image will move in the direction you press from the center of the image. A smaller image (thumbnail) of the whole guide image is displayed in the upper right part of the window. The area seen in the large display is identified in the thumbnail with a colored box.

Usage:
Pick AOI's by moving around the display of the guide image until a region is seen where a high-magnification image is required. Press on the 'Select Menu and position the appropriate magnification over the AOI. Repeat until all AOI's are identified. Then press "Get HM Image" to send the request to the other station.

3.4.9 "High Magnification" Window

Synopsis:

This window displays a high-magnification image that was send from another station for examination. The region in the guide image where this image is located can be seen in the thumbnail of the guide image in the upper right corner of the window. Information about the location and magnification used to create the high-magnification image are located in an information window below the image.

Operation:

"Play Message"
Will play a recorded sound message from the other station, if one was recorded. This information may tell you what the sender did to make the image and what he may want you to know about the image.

"Return"
Will take you back to the "Received Images" menu to pick another guide image to review.

"Record Message"
Will allow you to record a sound message associated with this high-magnification image. The message will be sent back to the other station requesting the examination and should contain the diagnosis and other relevant information concerning the slide.

"Next Location"
This will move the microscope to the next high-magnification location you previously requested (Loc. will increment by one up to the # of Loc's as displayed in the window below the image.) If this is the last location, the program will take you back to the "Received Images" menu to work on another guide image and associated high-magnification images set. If no other guide images have been received, the program will take you back to the main menu.

Usage:
Review this and other high-magnification images you previously requested for this guide image and make a diagnosis. The diagnosis can be written into a sound file by pressing the "Record Message" and it will be sent back to the other station.

3.4.10 "Stored Images" Window

Synopsis:
Allows the operator to view stored guide and high-magnification images.

Operation:

"View Guide Image"
Takes the selected guide image and displays it in Screen 11.

"View High-Magnification Image"
Takes the selected high-magnification image and displays it in Screen 12.

"Quit"
Takes you back to the main menu.

Usage:
When the screen is called up, a list of all the stored guide images will be displayed on the left window "Guide Images". If one is selected by pointing to the file name, a thumbnail of that guide is displayed in the "Guide Image" window. Also, any high-magnification image files that were stored that accompany that guide image will be listed in the right window "High Mag Images". If one of the high-magnification files is selected by pointing to the file name, a thumbnail of that high-magnification is displayed in the "High-Mag Image" window. Either the guide or high-magnification image can be displayed in a full screen by pressing either "View Guide Image" or "View High-Mag Image".

3.4.11 "Display of Stored Guide Image" Window

Synopsis:

This window displays the guide image that was stored from a previous examination. You can move the image around to see various parts of the guide image and play a sound message associated with the guide image.

Operation:

"Play Message"
Will play a recorded sound message, if one was recorded.

"Return"
Will take you back to the "Stored Images" menu to select other guide or high-magnification images.

"Guide Image"
The display of the guide image is only a small part of the full slide. To move the image around to see more of it, you can press on the guide image and the image will move in the direction you press from the center of the image. A smaller image (thumbnail) of the whole guide image is displayed in the upper right part of the window. The area seen in the large display is identified in the thumbnail with a colored box.

Usage:
Only used to review stored guide images.

3.5 Technical Specifications:

High-resolution image (guide image)

CCD linear array	Kodak KLI-4103
Resolution	4104 x 3 pixels, 12 uM
Clock rate	1.0 mHz
Acquire	2900 x 2048 x 24 bit color
Display	2900 x 2032 x 24 bit color
Acquisition time	8.5 seconds
Compression time	30 seconds
Compression ratio	variable through software
Illumination	preset

High-magnification image

CCD area array	Sony DXC-960MD-(3-chip color)
Resolution	768 x 494 pixels, 9 uM
Display	768 x 484 image points
Acquisition time	33 mSec
Compression ratio	variable through software
Field of view	match optical resolution

Microscope

X,Y stage	124mm x 55mm travel, computer control with manual override via touchscreen
Focus	computer control of fine focus with manual override via touchscreen
Magnification	computer control with manual override via touchscreen
Condenser	computer control
Illumination	computer control with manual adjustment of correct value
Objectives	2X, 4X, 10X, 20X, 40X, plan apochromat
Eyepieces	10X, 20mm FOV

Computer System

CPU	Pentium, dual 90 MHz Intel processor
Tower	Full sized tower configuration
Driver controller	SCSI onboard controller
Fixed storage	2.0 Gbyte
Removable storage	270 mByte removable disk (Syquest)
Monitor	1280 x 1024, 17 inch touchscreen
Floppy drive	1.44 MB, 3.5 inch
Keyboard	101 keyboard
Data compression	JPEG compression via software algorithm
Frame grabber	Matrox Magic/RGB for both cameras (video switcher)
Transmission card	ISDN with two lines of 56 Kbits each
Printer	N.A.
Memory	64 MB RAM
Cache	512 KB
Slots	3 slots PCI bus 5 slots EISA bus
Ports	2 serial RS-232; 1 parallel
Graphics accelerator	Matrox Magic/RGB, 1024 X 768 X 256 colors
Power supply	300 W
Operating system	Microsoft Windows NT 3.50
Software language	C++, v. 2.20

4. HOSPITAL INFORMATION SYSTEMS

The purpose of this section of is to review state-of-the-art Hospital Information Systems, both for the military and civilian arena.

4.1 Hospital Information Systems

The HIS (Hospital Information System):

- manages hospital finances and resources
- furnishes decision support[vii] through ad-hoc reporting
- automates office work at all levels
- tracks/manages patient data, care and billing
- establishes inter-communication and data-exchange between:

  other hospitals, insurance and billing agencies, clinics, laboratories, nursing stations, other information systems and data repositories, wired or wireless instrumentation and printers, technologists, and physicians in the hospital or elsewhere.

However, current HIS are not what would be termed state-of-the-art. As pointed out in the recent edition of the New York Times, describing a leading medical institution, namely, Stanford Medical Center (California), "...when new patients enter the hospital, their medical information is recorded and distributed much as it was 30 years ago: on paper.

"As primitive as that may sound Stanford is actually ahead of most hospitals in adopting new information technology. It has dozens of computer systems and has more than 70 clinics, and about three years ago the hospital embarked on an ambitious program to link them in a common system...

"...To Gerry Shebar, Associate Director of Information Management and Technology for Stanford Health Services, the program makes Stanford something of a leader. The health care industry, he said, is 'where banking was 10 years ago, and the airline industry was 15 years ago - pretty much paper-based.'

"...so the broad changes underway in health care are creating a huge market for new technology.

"Hospitals...have gone through waves of computer purchases. Many individual physicians have bought personal computers, if only to aid their staff in pursuing reimbursement from insurers and Medicare.

"But these computers have been unable to communicate with each other, so any given bit of medical or financial information remained accessible to only a few individuals. And that most critical of documents, the patient record, lived on in paper form."

Thus, the purpose of this Section is to review the state-of-the-art of HIS, both for the military (Subsection 4.5) and the civilian arena (Subsection 4.6). By means of this review Kensal staff, who are engaged in modernizing conventional microscopy for pathology, will be aware of

the environment in which such new devices must operate. This will have an impact on the structuring of the associated hardware and software systems.

4.1.1 HIS Modularization

Since more feature-sets and end-users have been added to the HIS domain over the years, the HIS has become increasingly modular with outgrowths in clinical and laboratory information systems (CIS/LIS). *Savings* through better organization, increased automation, and faster order/result turnaround are the compelling reasons for outgrowth and modularization. Each sub-IS module not only acts as a store-and-forward/fault-tolerant repository of data between the central HIS[viii], but also provides domain-centric ad-hoc feedback to inquiring administrators who must report to hospital enterprise leaders and government regulating agencies. The six related and overlapping systems[ix] within the health care field are:

- o Management Information Systems
- o Financial Information Systems
- o Telemedicine Information Systems
- o Knowledge Systems
- o Public Health Information Systems
- o Research Systems

4.1.2 HIS Security

Throughout the chain of hospital rank and command, layered need-to-know based security protects sensitive information within the HIS. While nursing stations can prepare work lists based on patient needs, they cannot view executive level reports. While the admissions-desk can see bed availability in real time, they can not change laboratory results. While physicians can submit pharmaceutical and laboratory orders, they cannot admit, discharge, or transfer (ADT) patients. Security is necessary to protect patient privacy, control efficiency and order as well as prevent accidents, fraud, or deliberate reprisals and sabotage against the hospital enterprise.

4.1.3 Strategic Planning

The HIS, when well implemented, can be used as a powerful management tool to guide decision making. The ideal HIS provides rich insight and decision support for the optimal financial structuring of the hospital.

In the near future, civilian HIS will likely be linked to a Community Health Information Network *(CHIN)* to help minimize costs within a group of collaborating health-care providers. In addition, research by the NII-HIN Consortium is underway to develop standards to provide transparent linkages between CHINs through a national information infrastructure. Finally, there is some movement towards a Global Heath Network which is focused on the critical role of prevention in reducing health care costs through rapid, accurate transmission of information. Other innovation includes computer-based order/results entry and point-of-care reporting.

"CHINs are community-wide electronic networks of health care providers, medical facilities, payers, pharmacies, and other health care support companies that allow the sharing of patient medical and financial data in a more efficient manner. CHINs can also support the sharing of radiological images and live telemedicine. A regional CHIN promises to improve the quality of patient care and lower the cost of health care in the community." Before a hospital can be integrated into a CHIN however, it must support a *Computer-based Patient Record* (CPR)[x] that can be transparently passed to other hospital HISs of likely dissimilar implementation. Recently, enormous industry and media attention have been focused on the CPR. Despite this, hospitals in

general are hesitating to implement a CPR due to cost and complexity. CHINs are thus just emerging in the health care industry but will play a significant role in the near future of health care.

4.1.4 Computer-based Patient Record (CPR)

The CPR concept is fundamentally a computer-stored collection of health information about one person linked by a person identifier. The CPR or the "electronic patient record" are terms used by vendors interchangeably but refer to different levels of computerization. Clarification regarding these levels has been outlined by the Medical Records Institute[xi] (MRI), founded on the principle that the future of health information technology lies in the successful creation and implementation of electronic health record systems. Although in fact *five levels* have been defined, only the first two levels have been achieved--levels 3 through 5 are not felt to be possible for some time. The five distinct levels of computerization for patient information systems has been outlined by MRI as follows:

- Level 1: *Automated Medical Records*
  Are paper-based medical records with as much as 50% of the printed content computer generated. Level 1 automation within the hospital environment is focused around the following functions:
  - ADT (Admission/Discharge/Transfer) systems
  - Improved capture of patient information through digital dictation systems
  - Patient accounting and its linkage to clinical information
  - Departmental systems (i.e., Radiology Information Systems, Laboratory Information Systems, Pharmacy Information Systems, etc.)
  - Order Entry/Results reporting

  Other innovation parallel to the paper-based medical record are nursing/bedside computing (discussed in section 1.5), implementation of an enterprise-wide master patient index, the linkage of various parts into an enterprise-wide network, the development of interface engines (discussed later), and imaging.

- Level 2: *Computerized Medical Record System*
  Level 1 automation does not solve the space shortage in record storage, nor create an electronically available record. A level 2 computerized-medical record system (or document imaging system) allows paper-based medical records to be created, then scanned, and indexed within a computer system with the same automation functions as level 1. Optical Character Recognition (OCR) or Intelligent Character Recognition (ICR) do not fit into level 2 automation since the scanned documents are stored on optical disks as unchangeable images, not ASCII-based data-sets. Level 2 is the only method in existence as of this writing to computerize the medical record in a paperless system.

- Level 3: *The Electronic Medical Record*
  The level 2 computerized medical record has basically the same structure as the level 1 paper-based medical record. The level 3 electronic medical record has the same scope of information in level 2 but the information is rearranged for computer use. While the level 1 paper-based records system is a passive storage device, level 3 can provide interactive aiding of the decision making process by knowledge coupling, providing decision support, and many other functions. Level 3 requires a secure enterprise-wide infrastructure for appropriate capture, process and storage of patient information.

- o Level 4: *Electronic Patient Record Systems (also called Computer-based Patient Record Systems)*
The patient record has a wider scope of information than the medical record. It combines several enterprise-based electronic medical records concerning one patient and assembles a record that goes beyond the enterprise-based retention period.

- o Level 5: *The Electronic Health Record*
The more comprehensive collection of an individual's health information is the level 5 electronic health record. It differs from the electronic patient record in the unlimited amount of health information captured by care givers regarding a person. It includes wellness information possibly captured by the individual or parents, therapists, etc., including data for example on behavioral activities such as smoking, exercising, dietary and drinking habits. The electronic health record is maintained through the cooperation between the individual who controls his or her health information, and the care giver.

4.1.5 Computer-based Point of Care

A recent HIS appendage and innovation for cost-savings *are computerized point-of-care systems*. HIS vendor CPSI markets a "Chart Cart"--a portable PC on a medicine-cabinet cart with a touch sensitive screen and bar code reader, all wirelessly connected to the HIS--which allows Nursing Services personnel to enter information into the HIS at the patient bedside. Clerical functions are automated and duplicate entry of information into nursing documents is eliminated. Charges for administered medication can be billed immediately by using the keyboard and bar code reader to scan the medication container.

MEDITECH also has a 14 ounce hand-held personal data assistant (PDA) for computer-based point of care. End users of this device are nurses, nurses aides, and therapists. The PDA holds data for 10-20 patients and keeps track of the "whereabouts of physicians". When a nurse's shift begins, the nurse downloads patient records into the PDA and then administers to the need of the patients. During the shift, the nurse can operate the PDA with one hand's thumb--to see orders and record results--while administering care with the other hand. After the shift, the data in the PDA is uploaded into the HIS.

4.1.6 Health Care Information System Priorities

Relatively new (and growing) in the HIS outgrowth are multimedia systems, wireless LAN technology, mobile Personal Data Assistants (PDAs), and telemedicine. The current priority however among health care providers today is "integrating systems across separate facilities", which outranks, "implementing a computer-based patient record, reengineering to a patient-centered computing environment, and incorporating wireless/portable devices".

In a January 1995 health-care survey of nearly 1000 respondents, "HIS budgets will grow at a healthy clip: more than 80% say HIS budgets are going up. Over half stated that budgets would increase 30% or more." The surveyors thought that these statistics reflected a growing concern to integrate health care networks. Thus, many existing HIS implementations still lack enterprise-wide integration.

4.1.7 Computer Imaging In HIS

Computer imaging is relatively new to the HIS. Several outgrowths are currently integrating images and text in stand-alone modules (i.e., radiology, paperless-office, and telemedicine modules). However, there is no standard way to integrate the text-based medical record and related digital image-based entities together for call-up throughout the HIS, much less

across hospitals or CHINs. Thus, a tremendous amount of work yet lies ahead to create, what might be coined as, the "Graphical Patient Record" (GPR).

While standards do not exist yet meshing text and large binary objects like images for HIS-wide access, Los Alamos National Laboratories (LANL) has recently announced TeleMed which contains an experimental GPR focused currently in teleradiology. TeleMed, based on a distributed national radiographic and patient record repository which could be located anywhere in the country, is designed to assist doctors in treatment planning through viewing patient treatment histories and associated radiographic data. This data can be viewed simultaneously by users at two or more distant locations for consultation with specialists in different fields. LANL claims that TeleMed "is the first to provide transparent access to patient record components over a WAN, building the complete patient record from various partial records and displaying that in an integrated manner to the healthcare provider."

Industry standards are needed for seamless integration of images throughout the HIS. Once again, no standard exists which integrates text and images across the entire HIS as of this writing; however there are several SDO's (Standards Developing Organizations)--who have good foundations and the technical resources--developing such a standard. (See below.)

4.1.8 Reports Available From The HIS

Available HIS reports are endless and their titles vary from vendor to vendor. Often, vendors will tailor report content and structure to the needs of the hospital. Thus, unlike the IRS or other government branch, there is little report standardization, except in the insurance billing modules and in the reports destined for accreditation overseers.

Often provided in the various HIS modules, is the ability to generate ad-hoc reports; thus, in addition to the "canned" reports. Reports of any content or structure can be generated through SQL (Structured Query Language) inquiries on a database. However, the HIS end-user must learn how to use the SQL interface and the semantics of the query language before useful reports can be generated.

As mentioned earlier, an extreme interest in moving to a paperless reporting mechanism has been manifest in many hospital enterprises, due to cost savings. Most of the HIS vendors are just now beginning to offer the "Level 2" document imaging ability. "Level 3" is highly desired, but requires physical and logical integration across disparate facilities and computer systems, with nearly a unique solution for each integration case. To understand the barriers to enterprise-wide electronic report exchange, the physical and logical architecture of the HIS will be discussed in the next section.

4.1.9 Standards

There are many standards groups who's specifications are being used to implement the HIS. At the *messaging level*--the level where HIS nodes exchange information related to the health-care industry--various standards groups, many driven by HIS vendor innovation, have been working together to build the expanding field of *medical informatics*. At the lower hardware level, IEEE, ISO, ITU-T (CCITT), ANSI, et al, have published networking specifications in circulation for years, used in HIS implementation. Newer negotiated-multiband technologies such as ATM[xii] (Asynchronous Transfer Mode) for information interoperability are also being used in some HIS implementations.

4.2. Medical Informatics

"Biomedical Informatics is an emerging discipline that has been defined as the study, invention, and implementation of structures and algorithms to improve communication, understanding and management of medical information. The end objective of biomedical informatics is the coalescing of data, knowledge, and the tools necessary to apply that data and knowledge in the decision-making process, at the time and place that a decision needs to be made. The focus on *the structures and algorithms necessary to manipulate the information separates* Biomedical Informatics from other medical disciplines where information content is the focus."

A medical informatics USENET newsgroup is open to the Internet public at:

sci.med.informatics

4.2.1 Medical Informatics Standards Groups

The term "standards"[xiii] includes standards developed by accredited standards organizations and other categories of organizations who are affecting, or working on, technical, procedural, and systems standards, guidelines, professional protocols, minimum requirements, as well as industry practices necessary to enable the computer-based record system of the future to function. From this perspective, there are seven categories of organizations involved in the process:

- Major standards organizations who develop application standards for health care
- Professional societies involved in standards creation
- Trade associations
- Government organizations
- Industry consortia
- National players
- Standards organizations for base standards

The Medical Records Institute[xiv] provides an *International Directory of Organizations: Standards and Developments in the Creation of Electronic Health Records* [xv] which lists over 160 different groups working on standards in health care throughout the world; outlining their current projects, publications and reports.

One of the largest components in the HIS standards work in progress is the design effort taking place to specify how digital messages should be exchanged between HIS computer systems and what they should contain. These messages encapsulate information ranging from ADT updates to lab-results data. The messaging structures implemented in HIS systems today are analogous to the different foreign languages and/or dialects spoken in various regions of the earth--from the global HIS market perspective, every vendor has its own unique standard or, more frequently, *interpretation* of a local recognized standard (i.e., HL7, discussed later). Since a substantial technical investment is required to enable one vendor--faced with appending modules on to HIS systems from other vendors--to speak all these languages and dialects, convergence to a common language--or messaging standard--is the drive behind the messaging SDOs today.

4.2.1.1 The Message Standards Developers Subcommittee (MSDS)

In 1991[xvi] there were at least six organizations developing health care messaging standards, of which three were accredited by the American National Standards Institute (ANSI). During that year, the European standards agencies asked ANSI to clarify with whom they could coordinate health informatics standards. As a result, ANSI formed the Health Informatics

Standards Planning Panel (HISPP) to coordinate the development of health informatics standards. HISPP's membership includes system vendors, professional organizations, Standards Developing Organizations (SDOs), and various users of standards.

In turn, HISPP formed a subcommittee of its members who were standards developing organizations. This is the Message Standards Developers Subcommittee (MSDS). The members of MSDS are SDOs developing health care message interchange standards. The objective of the MSDS is to achieve harmonization of the standards that SDOs develop.

Members of MSDS are:

- **ASTM**: American Society for Testing and Materials
- **DICOM**: Working groups of American College of Radiology (ACR) and National Electrical Manufacturers Association (NEMA)
- **HL7**: Health Level Seven
- **IEEE**: Institute of Electrical and Electronics Engineers Medical Data Interchange Working Group
- **NCPDP**: National Council of Prescription Drug Pharmacies
- **X12N**: Insurance Subcommittee of ASC X12

The MSDS formed the Joint Working Group for a Common Data Model (**JWG-CDM**) as an open standards effort to support the development of a common data model that can be shared by developers of health care informatics standards. The IEEE Committee has secretariat responsibility for the activities of the JWG-CDM. Thus, for all practical purposes, the IEEE Medical Data Interchange Working Group and the Joint Working Group for a Common Data Model are identical. The acronym JWG-CDM refers to these groups.

On June 6, 1994 the IEEE Standards Department made available the initial draft of the JWG-CDM standard as four postscript documents.

Duke University, North Carolina, maintains a repository for MSDS electronic files at:

(WWW) http://dumccss.mc.duke.edu/ftp/standards.html
(FTP) dumccss.mc.duke.edu

In addition, DICOM maintains electronic information at:

(FTP) xray.hmc.psu.edu

4.2.1.2 Health Level Seven (HL7) - Background

HL7 was founded in 1987 to develop standards for the electronic interchange of clinical, financial and administrative information among independent health care oriented computer systems; e.g., hospital information systems, clinical laboratory systems, enterprise systems and pharmacy systems.

In the last three years, its membership has tripled to over 1,400 hospital, professional society, health care industry, and individual members including almost all of the major health care systems consultants and vendors. The HL7 standard is supported by most system vendors and used in the majority of large U.S. hospitals today. It is also used in Australia, Austria, Germany, Holland, Israel, Japan, New Zealand and the United Kingdom.

Currently, HL7 does not support images but is working with the ACR to merge the DICOM standard with HL7 for image support. As of this writing, the current version of HL7 is v2.2; image support will not be available until v3.0, which is due in 1996.

HL7 minutes, standard drafts, and sample source-code are available through Internet FTP servers on [dumccss.mc.duke.edu], WWW URL:

http://dumccss.mc.duke.edu/ftp/standards.html

Also supported is a discussion group on the HL7@Virginia.EDU list server.

Virtually all HIS vendors are HL7-compliant and most of the world, including the military, is merging their HIS systems and sub modules into this standard. However, each vendor's implementation of HL7 is somewhat different--a unique interpretation. Thus, while HL7 provides a strong measure of order to the messaging dilemma between HIS systems and sub-modules, it doesn't eradicate all communication problems. Interfacing two HL7-compliant systems, for example, requires much work on a technical level.

4.2.2 Data Interface Engines

Because of the complexity involved in interfacing modules to HIS systems, each with its own interpretation of a recognized messaging standard, many system integrators are turning to "data interface engines" to simplify the process.

Interface engines (IEs) are a complex middleware technology also known as integration engines, interface hubs, and application interface gateways. Typically, an IE is a separate computer which acts to transform[xvii] messages between other computer systems and their applications. These disparate applications must have the ability to exchange messages, for example through a messaging application programming interface (API).

In the hospital environment, such IEs are used between HIS modules (i.e., the ADT module and the Radiology Module) perhaps purchased through different vendors[xviii] with different hardware/software implementations. The benefits of using an IE include, 1) simplified HIS interface development since the IE is a tool-set designed specifically for that purpose, 2) centralized interface management capabilities (i.e., starting, stopping, monitoring, trouble-shooting), 3) superiority over point-to-point (PTP) interfaces since complexity is reduced through use of the centralized IE hub (i.e., if 5 different systems requiring bilateral interfaces need to interoperate with each other, 20 PTPs are needed, while only 10 interfaces are needed to an IE-based implementation--adding another node to the former requires 10 more PTPs while the later only 2 interfaces), 4) possible reduced costs for IE-based interface implementation when compared to paying application vendors for installing a PTP-based interface, 5) the ability to populate clinical data repositories or data warehouses by routing data from messages exchanged between other applications, 6) an established CHIN entry-point for an organization.

IEs ideally send messages following the HL7 standards. However, some EDI[xix] transaction sets, and ASTM[xx] messaging standards are also used. Further informaiton on networks for HIS is given in Appendix A.

4.3 U. S. Military HIS

The US military has standardized its HIS installations around the world through the Composite Health Care System (CHCS), developed by SAIC. (The Veterans Administration uses the Decentralized Hospital Computer Program (DHCP), developed for the VA hospitals.

4.3.1 Science Applications International Corporation (SAIC) Overview

Science Applications International Corporation (SAIC), a privately owned defense contracting company headquarted in San Diego[xxi] with 19,000+ employees nationwide, enjoyed $1.9 billion in revenues for FY94 with 86% coming from the federal government. Outside the national security community, few have heard of SAIC.

Founded in 1969 by J. Robert Beyster[xxii], SAIC's principal product is brainpower. It acts as a systems integrator to design solutions to the government's toughest technology problems. SAIC's past projects include designing one of the early "star wars" antimissile defenses, the FBI's computerized fingerprint-identification system, and plant monitoring equipment for power plants. SAIC has also designed and built the <u>largest hospital information system in the world</u> as well as the <u>largest medical telecommunications system in the United States</u>[xxiii]. Recently, in a <u>telemedicine</u> experiment, SAIC helped link physicians aboard a hospital ship off the coast of Haiti with major U.S. military hospitals. As a result, the ship's doctors were able to give U.S. soldiers better medical treatment. Finally, SAIC's newest DoD health care contract involves building <u>community networks</u> linking military medical facilities with civilian providers and VA medical centers.

4.3.2 SAIC's Composite Health Care System (CHCS) Program

The mission of the Department of Defense (DoD) health care system[xxiv] includes maintaining the health status of the military force (including family members and retirees and their family members) by providing cost-effective, high quality inpatient and outpatient medical and dental care and maintaining medical readiness to support mobilization. It includes all inpatient medical facilities and all significant outpatient facilities, to include care delivery in the military theater and veterinary services.

Medical data processing capabilities are being acquired to assist the health care providers and administrators in the management and delivery of quality care to the patient population served within the DoD health care system. A flexible solution is being provided in medical data processing capabilities for all DoD medical treatment facilities (MTFs). Both large and small MTFs will be supported via a standard Composite Health Care System (CHCS). The architecture design involves an integrated hardware and software solution, fully scaleable to the range of DoD medical facilities, from small stand-alone facilities to large regions and outpatient catchment areas (OCAs).

4.3.3 Description of CHCS

The Composite Health Care System is funded through a $1.1 billion contract, SAIC's largest program. CHCS fundamentally is an automated network handling military health care, including patient scheduling, admissions, prescriptions, lab tests, and record keeping and was developed from close cooperation between the Pentagon and SAIC. CHCS has been installed in over 600 medical facilities worldwide and is also used in mobile units such as the one deployed in Gutanamo Bay, Cuba (discussed later in section 5.6).

The CHCS is a fully integrated, automated information system that supports the administration and delivery of health care in MTFs. The integration of this information system means that all data about a patient need only be entered once, authorized users have access through the system to all data needed to perform their functions, and all functions are available to authorized users.

The current functional baseline supports the following areas:

Registration of patients into one database
Eligibility (DEERS) checking
Patient Appointment and Scheduling (PAS)
Patient Administration (PAD)
Nursing
Inpatient and Outpatient Order Entry and Results Reporting
Pharmacy
Laboratory
Radiology
Clinical Dietetics
Quality Assurance
Medical Services Accounting
Electronic Mail
Integration with other standard DoD and Military Department automated information systems

Pre-Planned Product Improvement (P3I) for CHCS includes those data specific to individual patients that are most cost-efficiently captured and stored along with all other elements of the patient's electronic record, but that are not part of the original functional baseline of CHCS. These data derive largely from the Managed Care Program (MCP), including such things as other health insurance, enrollment data, and third party collection support. The life cycle cost and benefits of automated information system (AIS) support for these data will be derived from functional economic analysis.

The major technical design objectives are:
Ease of use;
Minimal program risk;
Ease of maintenance;
Flexible system configurations to support future enhancements;
Ease of operation;
Appropriate system security;
Scalability (small to large);
Reliability and Availability greater than 99%, and;
Adequate performance from user perspective.

4.3.4 History Of CHCS

Feb 79
MILESTONE 0: Mission Element Needs Statement (MENS) approved

1981
Initial Operating Capability (IOCs) systems (TRIPHARM, TRILAB, TRIRAD, and TRIPAS) deployed until an "integrated" system could be developed
Contracts let would expire in 8 years

Dec 84
MILESTONE 1: Major Automated Information Systems Review Council (MAISRC) approved CHCS requirements

Sep 86
Four (4) vendors selected to test CHCS software development and deployment with one of the vendors having to use the VA-DHCP software
Baxter Travenol to Sheppard AFB, TX
McDonald Douglas to Camp Lejeune, NC

SAIC (with VA-DHCP) to Ft Knox, KY
Technicon Data Systems (TDS) to Jacksonville, FL
Tri-Service Micro Pharmacy Systems (TMPS) deployed as part of TRIPHARM contract

Mar 87
Development/deployment to test sites started. Vendors had one year to develop and deploy CHCS to their respective sites TDS dropped out of the competition

Mar 88
MILESTONE II: SAIC selected as prime vendor for CHCS

Jul 88
SAIC to deploy CHCS to eight (8) additional "beta" sites the current software running at

Ft. Knox
Development of additional functionality would continue at Ft Knox. The beta sites, however, demanded any new functionality that was developed at Ft Knox be provided to them.
Milestone III set for Jul 89
New beta sites
- Air Force
  - Keesler AFB, MS
  - Sheppard AFB, TX
  - Eglin AFB, FL
- Navy
  - Camp Lejeune, NC
  - Jacksonville, FL
  - Charleston NB, SC
- Army
  - Eisenhouer, GA
  - Nurenberg, GE

Nov 88
SAIC was unable to deploy to new sites and deploy new functionality at the same time. Startup problems caused the CHCS program to be "frozen" for 3 months until software got back under control. This led to software version 2.0 or better know as the "Knox Baseline"
Milestone III delayed to Dec 89

Jan 89
Deployment restarted at beta sites

Oct 89
In Process Review (IPR) by MAISRC
- GAO stated that the current beta sites did not represent the largest nor smallest DoD hospitals and did not test the capability of CHCS to support readiness
  - Walter Reed and Bethesda added as representatives of the largest DoD hospitals
  - Shaw added as representative of DoD smallest hospitals
  - Carswell added to test readiness capability
- With multiple services using CHCS on a single host like in Hawaii, DoD was required to develop software to support "divided work center" areas. The divided work center (DWC) software would be ready for Version 3.0
- Tripler and Hawaii added as alpha site to test DWC software

Milestone III delayed to Oct 90
Support contracts for IOC systems expired Sep 89

1990

- Funding became a problem
- Carswell and Bethesda were removed as beta test sites
- Contracts for IOC systems had to be extended
  - As an interim, only enough CHCS functionality would be deployed to replace the IOCs (IOC-R Program). Software Version 4.0 would provide enough capability to replace the IOCs
  - Congressional language for FY91 for IOC-R
    - "The Secretary may not authorize the use of CHCS to replace an IOC in any MTF that is not involved in the OT&E phase....until the Secretary certifies to the Committees on Armed Forces of the Senate and House of Representatives that the use of CHCS is the most effective method for maintaining automated operations at the facility"
  - Congressional language defining certification
    - "Software to be used in a given MTF must be successfully tested at a representative MTFs"
    - "Software must be stable and all critical system incidents must be eliminated"
    - "The hardware must be properly sized at that facility to ensure adequate capacity when full configuration is installed"
    - "Installation of the IOC-R system must not adversely effect SAIC's capability for continuing OT&E"
  - Malcolm Grow added as test site for IOC-R software
- CHCS sites have performance problem
- MAISRC approved the Standard Appointing and Scheduling System (SASS) deployment
- Milestone III delayed to Mar 92

1991

- AQCESS required upgrade or replacement. Software Version 4.1 would replace AQCESS (AQCESS-R Program)
- IOC-R Program canceled
- Disk space and performance become a problem. GAO stated the two issues are related and demand CHCS be able to archive (and retrieve) data off the system
- For Mar 92 Milestone III decision, MAISRC and GAO agreed on the following
  - Milestone III was split into IIIA (Outpatient functions) and IIIB (Inpatient Order Entry (IPOE))
  - Early Operational Assessment of Version 4.1 (look in the SAIC lab only)
    - AQCESS-R software
    - Archive/Retrieve capability
  - Software Version 4.0 at six of the beta test sites
  - Charleston Performance Report (Charleston was the first site to upgrade the MicroVAX 4200s to VAX 6400s)

1992

- PAS becomes critical at some of the IOC sites
- Since PAS is the most stable CHCS software and most likely the first module to be "certified", the MAISRC approved deployment of PAS only at the following AF PAS Certification Sites
  - Travis
  - Scott
  - Offutt
- IOCs become more critical

MAISRC approved CHCS 19 May 92 "pending OT&E report". OT&E report did not go out until 15 Nov 92. Congress had 60 days to approve. If no action was taken, approval would be automatic

1993

- CHCS passed Milestone IIIA 15 Jan 93
- In Feb 1993, CHCS underwent a MAISRC IPR with the following directions
  - IOCs must be replaced by Sep 94 (For the AF, this was to be accomplished by MEDSITE)
  - TMPS would be replaced by Sep 94
    - TMPS was to be replaced immediately using CHCS on a single personal computer (PC) running under the DOS operating system. The system became known as PC-SAPH (Stand Alone Pharmacy)
  - AQCESS must be replaced by Sep 95
    - This would be available later in 1993 using multiple PCs running under the UNIX operating system. The system became know as PC-CHCS.
  - Managed Care Program (MCP) software must be deployed to the right base at the right time
- In Aug 93, TMPS-R and AQCESS-R programs were combined into a single program

1994

- In Jan 94, PC-CHCS was deployed to 6 alpha sites, 2 from each service
- In Jun 94, PC-CHCS was approved by OT&E

4.3.5 MEDSITE

MEDSITE is an acronym for **MED**ical **S**ystems **I**mplementation and **T**raining. Approved by the Air Force Surgeon General in March 1993, MEDSITE's mission was to deploy CHCS to those Medical Treatment Facilities (MTFs) which had existing Initial Operating Capability (IOC) systems (TRIPHARM, TRIRAD, TRILAB, TRIPAS).

When PC-CHCS was approved for accelerated deployment to all other MTFs, Lt Gen Sloan approved a ramp up of MEDSITE and Standard Systems Center (SSC/SBM) to deploy CHCS Patient Appointing and Scheduling (PAS), Patient Administration (PAD), Managed Care Program (MCP) and Pharmacy (PHR).

SSC/SBM hired 4 of 38 needed term employees to deploy PC-CHCS to 29 MTFs in eastern CONUS/USAFE. MEDSITE hired 54 term employees to deploy PC-CHCS to 30 MTFs in western CONUS and PACAF, to manage the PC-CHCS project and to operate an AF CHCS Support Center.

MEDSITE currently maintains a software team which develops interfaces between CHCS and other various medical information systems, as well as report generators and other specific modules. Some interfaces are developed as a final deployable product while others are developed as a prototype effort to provide a proof of concept and provide a better understanding of the level of effort required to develop a fully functional interface for the system in question. The team also develops hard coded report modules in situations where using a generic ad hoc report generating tool is ill suited to the task either because of complexity or performance.

Current MEDSITE Team Members:

Maj. Ray Bender, Director

E-mail: ray.bender@xmail.ha.osd.mil

Mr. Gregory Zymbaluk, Computer Sciences Corporation, Programmer

E-mail: gregory.zymbaluk@xmail.ha.osd.mil

Mr. Eugene Gonzales, Computer Sciences Corporation, Programmer

E-mail: eugene.gonzales@xmail.ha.osd.mil

Team Member Emeritus:

Capt. Loretta Hagen, Chief (former)

MEDSITE maintains WWW pages at URL:

http://bender.brooks.af.mil/

This server has descriptions and M source code of the public domain software that is currently available from MEDSITE, and to be developed in the future. Some of the interfaces that have been developed are:

Telephone Refill
TransLux DataWall
Pyxis Medstation ADT
Provider Workstation Results Retrieval
TRAC2ES Patient Movement Request
MICROMEDEX

Some of the report generators that have been developed are:

Pharmacy Cost Reports
Medicare Eligible Cost Reports

MEDSITE's deployable systems have been installed at:

Guantanamo Bay, Cuba - (Operation Sea Signal)
Zagreb, Croatia - (UN Protective Forces)

MEDSITE's required future work includes:

Deploy CHCS LAB to all AF MTFs by Dec 95
Deploy CHCS RAD and Order Entry by Dec 96
Support training for software upgrades for existing MTFs

Future work for MEDSITE may also involve becoming or forming an executive agent for the Consolidated Medical Systems Support Center (COMSSC) .

4.3.6 Case Study Of Remote USAFB Chcs Site: Guantanamo Bay, Cuba

This section will provide excerpts from a May, 1995 USAF "After Action Report" [xxv] which describes the humanitarian-mission/medical-effort carried out recently in Cuba, code-named

"Operation Sea Signal". These excerpts will serve to explain how CHCS was deployed in a mobile context and what the various camp implementation issues were for that context.

4.3.6.1 Excerpts From Executive Summary of Operation Sea Signal

As part of Operation Sea Signal humanitarian mission, the Joint Task Force (JTF) 160 Surgeon General (SG) was responsible for the care and support of the 21,000 Cuban migrants and approximately 500 Haitian migrants housed at the Guantanamo (GTMO) Bay encampments. Specifically, the medical care for the migrants was provided by the 6th and 59th Air Transportable Hospitals (ATHs). There was a wide range of medical services provided by these ATHs.

There was little automation deployed with the 6th and 59th ATHs. The requirements for basic medical automation in an ATH are the same as any fixed medical treatment facility - pharmacy, lab, radiology, results retrieval, patient registration and electronic mail. The purpose of this deployment was to support these basic requirements as well as validate new requirements specific to a deployed unit.

The major deficit in GTMO and within the ATHs was the lack of any type of computer/communications infrastructure. Naval Base (NAVBAS) GTMO had a wide area network (WAN) but the ATHs were not located in any area easily linked to this WAN. Secondly, the telephone infrastructure was saturated. Within the ATH, administrative duties were accomplished through the use of personal laptop computers that people had brought from home stations. After 11 months of use, they were beginning to break down and there was much concern about replacements. Telephones were limited to "field" phones linked by 4-wire tactical lines. At the 6th, there was not any link to electronic mail within the ATH or a link into the Internet. At the 59th, located across the street from the Camp Bulkeley J-6 (USMC), they had found a means to link up to the J-6 Banyan Vines server through tactical wire to provide them with access to e-mail at home. The 59th had no connectivity within the ATH. The pharmacy at the 6th ATH had brought Z-248 Tri-Service Micro Pharmacy System (TMPS) but they continued to have breakdowns. The 59th ATH did not have TMPS but did have the capability to use a personal computer (Z-248) with Pharmacy Label Producing Software (PHLAPS) for printing prepack labels.

MEDSITE's deployment of the Composite Health Care System (CHCS) to GTMO Bay was prompted by a request from the pharmacist assigned to the 6th ATH. After receiving approval from the ATH Commander, the JTF/SG, USACOM/SG, and the AF/SG, MEDSITE put together an DEC Alpha AXP capable of supporting a minimum of 25 concurrent users and enough disk storage for one year of on-line data. The system was installed in the 6th ATH with plans to tie all medical activities together.

4.3.6.2 Deployment Strategy/System Configuration

MEDSITE deployed a DEC Alpha (AXP) 3000/300 with CHCS Version 4.2/MU2 software. Peripheral hardware included DEC VT 320s, LA75 text printers, and Data South 300 XL label printers. Connectivity was via Local Area Transport (LAT) using DECServer 300s. Connectivity to outside locations was accomplished by connecting line drivers and bridges/routers through phone or tactical lines. Other specifics for hardware are listed below:

Product or Function	**Item**
CPU	DEC Alpha AXP 3000/300 RISC based 125mHz
DEC	1.5 GB DAT backup storage tape
StorageWorks	1.2 GB Disk Drive

	CD-ROM
Memory	64 Megabytes RAM
Disk Storage	20 Gigabytes (10 - 2.01GB disk drives)
Backup	Disk to Tape
	Disk to Disk
Operating System	OpenVMS Version 6.1
Software	DSM Version 6.3d
	CHCS Version 4.2/MU2
Other	TGV Multinet
	PWS/TRAC2ES Interface software

4.3.6.3 Communications

The AXP only has a 10BaseT connector and the DECServer only has a 10Base2 connector. A Boca Hub with a 10BaseT and 10Base2 was used to connect the AXP with the DECServer 300s. Running 6-wire unshielded twisted pair within the 6th ATH, VTs and printers were connected to DECServer 300s.

A link between 59th to JTFJ6 already existed. The Camp Bulkeley J6 (USMC) had connectivity between their Banyan Vines server and the JTFJ6 Banyan Vines server in the Pink Palace. A tactical line from the 59th ATH had been run across the street to the Camp Bulkeley J6. Since the JTFJ6 at the Pink Palace was linked to the Internet, both the 59th and the Bulkeley J6 were linked to the internet. The goal was to link the 6th ATH into the same Banyan Vines server at the Pink Palace so we could access either the internet or the 59th ATH. If the NAVHOS[xxvi] GTMO had access to the internet then we could theoretically access them once we were on the internet.

Linking the 6th ATH to the JTFJ6 Banyan Server. The Navy Communication Detachment (NAVCOMMDET) at GTMO provided two cable pair that we used to attach two AT&T 3510 line drivers and two DECrouter 90T1 bridge/routers. One end was attached to CHCS via the Boca Hub while the other router and modem were attached to the JTFJ6 Banyan Vine server. We had continuous problems with keeping the link up between the two modems. When the link was up we were able to telnet to the Banyan router and get to the internet.

Linking the 6th ATH to Camp Clinics (first is Lima/Mike camp) and the 59th to Camp Clinics (first is Echo/Foxtrot camp). Although two Codex 3500 line drivers were taken to connect Lima/Mike with the 6th, they were never tested because the lack of cable pair or commercial phone lines going to these clinics. A link in the future would require some type of wireless technology. LCDR Tillery and LT Welch visited from Naval Medical Information Management Center (NMIMC), they had discussed the installation of a cell on one of the hills and using cellular phones/modems to hook up the ATHs with their outlying clinics.

Connect to NAVHOS GTMO. The 6th ATH and the NAVHOS were both able to provide a single phone number that allowed modem access between the two facilities. Although not very fast we were able to link the NAVHOS Lab to CHCS using a pair of 2400 baud modems. Although we had taken 9600 baud modems we were unable to get the DECServer 300 to talk to them. Between the Pink Palace, Deer Point, and NAVHOS there is a clear line of site which is less than 4 miles total distance. Wireless technology could be used in the future.

4.3.6.4 CHCS Advantages

Results that were being recorded on separate log sheets and log books can now be found and printed in a collated report in less than five minutes compared to 20 minutes or more without CHCS. All specimens entered into CHCS were immediately accompanied by an audit trail providing positive specimen tracking. In addition special reports such as the Pending Lists, Overdue Procedure Reports, and Uncertified Results Report provided lab management an easy way to monitor the status of any test and take corrective actions to ensure results are returned in a expeditious manner not lost in a mountain of loose papers. Results were accessed from anywhere in the ATH there is a terminal, not just at the laboratory. This reduced the amount of time wasted walking to the lab to research what happened to a result. Electronic mail was used to pass information on protocol changes to different shifts, easing dissemination of critical operating policies.

4.3.6.5 CHCS In Emergency Unit

A CHCS terminal was placed in the Triage area (open tent adjacent to ER). This allowed the ER tech to triage the patient, take vitals, and print the 558 to the main ER. Changes were made to CHCS to allow the triage technician to enter directly into CHCS the patient's vital signs and to add comments he wanted to pass on to the ER. Once complete the 558 was printed on the ER printer. The bottom of the 558 was also changed to allow the understanding statement to print in Creole or Spanish.

An Information Desk Display was added to the Emergency Room main menu to allow for easy and fast look up of admitted patients.

4.3.6.6 DMPITS Database Conversion

Patient tracking was a problem at the ATHs. Some sections were using the US Atlantic Command (USACOM) developed Defense Mass Population Identification and Tracking System (DMPITS). There were multiple problems identified with DMPITS: (1) lack of confidence in the data accuracy because registration information was not verified at the time enrollment; (2) lack of devices in each section (many of the devices were broken and did not work); and (3) the DMPITS was not on a network, leaving each section to build their database. DMPITS was updated manually once per week based on data provided to a central location. CHCS would provide a means for accurately tracking patients through the ATH as all would use one central patient database.

The DMPITS office provided a DOS "flat file" containing the Name, DMPITS Number, Date-of-birth, Sex, Camp, Tent, and Bed. This file was transferred to the Alpha using a laptop computer. A conversion program was written in MUMPS to read the file and insert the data elements into the CHCS database providing pre-registration for all migrants.

4.3.6.7 After Action Conclusions

The DEC Alpha proved to be the ideal platform for simplified system management required for a deployed system. A single CPU system eliminated the problems with database synchronization and greatly simplified back-up procedures. The performance was excellent and better than expected. Any deployable system should be fully scaleable if future upgrades become necessary. Finally, the OpenVMS operating system was very robust and tolerant of unexpected "crashes" that are often a fact of life when operating in a tent environment operating off generator power. All the ATH components (CPU, DECServers, VT 320s, LA 75s) were configured at MEDSITE and tested for compatibility and reliability prior to deployment. This part of the deployment went smoothly and as predicted. However, the remote communication solutions between all the medical facilities at GTMO were not tested because the availability of the type of physical wire was unknown. In the future one needs to know the location of the nearest Wide Area Network (WAN) connection and the locations of any remote sites that will be connected to the CPU. The distances from the CPU to these locations must be known as this will drive the

communication solutions. Based on this information, the team should deploy with one or more solutions for each type of remote connections. The deployment to GTMO was very successful. The ability to get daily A&D reports; the ability to track the pregnant migrant women by camp, DMPITS number, and EDC; the ability to better track and monitor drug distribution, whether by prescription or by bulk issue; the ability to quickly send panic lab values directly to the clinic or ER; and the ability to register and track patients all improved the efficiency and quality of care being given by the 6th ATH. From this test deployment, many lessons were learned regarding the flexibility of CHCS and the flexibility required to support both humanitarian as well as wartime missions. These lessons will be used to better train our people for future deployments.

4.4 Civilian HIS Vendors (HBOC, SMS, MEDITECH, KEAN, CERNER)

HIS Vendor/Rank		No. of Installed Units	Price Range Per Install
1)	HBO & C	2,600[xxvii]	$700 K to $1 Million[xxviii]
2)	SMS	-	-
3)	MEDITECH	700+	-
4)	KEANE	945	-
5)	CERNER	-	-

Of the above five, Kensal Corporation has received literature from HBO&C, SMS, MEDITECH, and KEAN. A brief overview of these firms is presented next.

4.4.1 HBO & Company (HBOC)

HBOC
301 Perimeter Center North
Atlanta, Georgia 30346
Phone (404) 393-6000
FAX (404) 393-6092

Literature Received:
- Corporate Overview . . . . No
- List of HIS Modules . . . . Yes
- List of Services . . . . Yes, Network solutions
- Complete description of each module . . . . No
- Innovative Peripherals . . . . No
- Third-party reviews . . . . No

4.4.1.1 Overview

HBOC is a healthcare information solutions company that provides information systems and technology for the health enterprise--hospitals, integrated delivery networks and managed care organizations. HBOC claims to offer products and services to meet virtually every need the enterprise has for information, whether patient care, clinical, financial or strategic management.

HBOC markets local, metropolitan and wide area network services; HBOC's client/server-based Pathways 2000 suite of applications provide key elements for integrating and uniting providers across the continuum of care and establish the infrastructure necessary for a lifelong patient record. Its hospital-based STAR, Series and HealthQuest transaction systems and TRENDSTAR decision support system--along with the clinician-focused Pathways 2000 products--help improve the delivery of health services to an entire community. The Pathways 2000 resource scheduling and managed care solutions and QUANTUM enterprise information system support the

critical business functions necessary to manage today's emerging health networks. In addition, agreements and alliances with business partners allow HBOC to offer a broad variety of complimentary applications and technology, such as physician practice management system.

HBOC wraps these products with such services as planning, implementation and support, plus education and training. HBOC also offers a range of outsourcing services that includes strategic information systems planning, data center operations, receivables management, business office administration and major system conversions.

4.4.1.2 HBOC's Network Solutions

HBOC has noted that healthcare is drastically changing in the way it conducts its business. Fee-for-service is giving way to managed care and competition. Stand-alone hospitals are being incorporated into health enterprises. Wellness is being measured by outcomes rather than amounts of care and patient chart size by transmission time rather than page count.

With such change, HBOC is attempting to address the following information requirement issues: 1) How do organizations share information among the many new players in a managed care environment? 2) How do they provide meaningful information for universal access throughout the facility? 3) How do separate organizations exchange the information required for a true computer-based patient record? 4) And how does any healthcare entity avoid system obsolescence in a technological environment that's advancing exponentially? 5) How do organizations build an information infrastructure to support a rapidly and constantly changing environment?

HBOC has formed "HBO & Company's Connect Technology Group" (CTG) to address the aforementioned issues based upon the conviction that retrieving, integrating and presenting information from disparate sources to an expanding variety of users will become critical in the new world of healthcare--and that networks will make these tasks possible. CTG has more than 20 years of healthcare industry knowledge, more than 100 healthcare network installations, advanced networking expertise and "proven experience" in providing information.

4.4.1.2.1 HBOC's Method and Approach to Network Solutions

HBO&C claims that is the only major HIS vendor able to provide "one-stop shopping" for hardware, software, information networks and maintenance. CTG offers complete network installation, including a variety of telecommunications services:

- **Requirements Definition:** HBOC performs a comprehensive analysis of the hospital's short-term and long-term communications needs.
- **Functional Design:** Based on information gathered during requirements definition, HBOC recommends LAN media and hardware and software configurations.
- **Final Design, Installation Planning and Project Planning:** HBOC completes the preliminary design fro the cabling system and develops and installation timetable and detailed installation plan.
- **Procurement and Materials Management:** HBOC orders LAN hardware, software and cabling components and carefully tracks these orders to ensure timely delivery.
- **Project Management and Installation:** HBOC monitors and periodically reports on all installation work to ensure timely and proper installation and

configuration.

- **Testing and Quality Assurance:** The CTG project management and installation team verifies all aspects of the LAN implementation including user training and satisfaction.

- **Post-Installation Operation Management:** Customers may contract for additional consultation, support, management or disaster recovery planning services.

HBOC's health enterprise networking services ranges from LAN to MAN to WAN, with the WAN applications still under development:

- **LAN (Local Area Network)**
  - Patient-Focused Stations
  - Outpatient Surgery
  - Patient Accounting
  - Materials Management
  - Medical Records
  - Physician Offices
  - Radiology
  - Pharmacy
  - Laboratory
  - Emergency Room
  - Point of Care
  - Business Office
  - Executive Decision-Makers
  - Wellness Centers
  - Technology Management Systems
  - Community Network Access

- **MAN (Metropolitan Area Network)**
  - Payers
  - PPOs, HMOs, PHOs
  - Alliances, AHPs
  - Employers
  - Physician Offices
  - Home Health
  - Consultants
  - Banks
  - Local Governments
  - Skilled Nursing Facilities
  - Referrals
  - Medical Literature Databases
  - National Network Access

- **WAN (Wide Area Network)**
  - Clearinghouse
  - Federal Government
  - Employers
  - State Governments
  - Payers
  - Clinical Databases
  - Banks

- o Financial Databases
- o National Diagnostic Centers
- o Computer-Based Patient Record

4.4.1.3 HBOC HIS Modules

HBOC markets two HIS systems under the titles STAR Solutions and Series Solutions with apparently few differences in the literature other than different platform implementation options and minor variance in offered module types. Since the STAR system is the newer, an overview of its major modules only is presented below.

The STAR Modules share a single logical database with immediate access to authorized users from any workstation on the network. Comprehensive reporting tools are provide by the STAR KB_SQL report writer.

4.4.1.3.1 STAR Patient Care

This is the patient-centric hub of the HIS network. STAR Patient Care is where vital patient information is entered, maintained, tracked and disseminated throughout all departments.

Application Software:

- o Patient Processing
- o Patient & Resource Scheduling
- o Nursing
- o Order Management
- o Scheduling Departmental Profiling
- o Physician View

4.4.1.3.2 STAR Radiology

STAR provides a computerized clinical and administrative radiology information system that serves the areas of diagnostic radiology, nuclear medicine, ultrasound, magnetic resonance imaging, special procedures, computed tomography, mammography and radiation therapy.

Application Software:

- o Patient & Resource Scheduling
- o Order Management
- o Exam Resulting/Reporting
- o Film/File Room Management
- o Activity Tracking
- o Report Review Process
- o Administrative/Management Reports
- o Historical Patient Index
- o Quality Assurance

4.4.1.3.3 Pharmacy

A pharmacy system that addresses both inpatient and ambulatory areas. HBOC claims that it opens new avenues of communication, enhances teamwork and streamlines operations.

Application Software:

- o Profile Management
- o Order Entry
- o Dispensing Management

- o Clinical Services
- o Inventory & Purchasing Management
- o Productivity & Management
- o Reporting
- o Formulary Maintenance

4.4.1.3.4 Laboratory

The goal of this system is to ensure quality of outcome through far-reaching communication, data integrity and management control. Extensive quality control features help ensure the reliability of test results and the appropriateness of care delivered.

Application Software:

- o Order Management
- o Test Processing
- o Patient Inquiry
- o Patient Result Reports
- o Quality Control
- o Administrative/Management Reports
- o Surgical Pathology
- o Advanced Microbiology
- o Advanced Blood Bank
- o Contract Management

4.4.1.3.5 Financials

An executive management tools and general financial applications module.

Application Software:

- o Patient Guarantor Accounting (Data Collection, Billing, Insurance Follow-up/Collections, Account Management, Third-Party Logs)
- o General Ledger
- o Accounts Payable
- o Materials Management
- o Human Resource Management
- o Medical Records

4.4.1.3.6 TRENDSTAR

A decision support system which offers "point and click" access to information, analytical functions, reporting and presentation tools.

Trendstar complements HBOC's STAR and HealthQuest financial and clinical products by providing managed care contracting and monitoring, quality and case management, budgeting, forecasting and strategic planning solutions to support enterprise-wide decision-making.

Application Software:

- o Hospital Systems Library
- o Clinical Cost Accounting
- o Contract Payment Advisor
- o Management Cost Accounting
- o Marketing Systems Library
- o Resource Utilization Analysts
- o EpiTREND Reporting System

- TRENDPATH

4.4.1.3.7 QUANTUM

An executive information system (EIS) that "empowers executives by pulling together up-to-the-minute information from all the key areas of the healthcare enterprise." This digital entry allows a user to monitor operating targets, critical success factors, market trends or daily events.

4.4.1.3.8 Hardware and Operating Systems

STAR Solutions:
- Hewlett-Packard UNIX
- Data General UNIX and AOS/VS II
- Digital Equipment Corporation's VMS
- IBM RISC System/6000 AIX

SERIES Solutions:
- IBM AS/4000 with OS/400
- RISC System/6000 with AIX

4.4.2 SMS (Shared Medical Systems Corporation)

SMS
51 Valley Stream Parkway
Malvern, PA 19355-1406
Phone: (610) 219-6300
FAX: (610) 219-3124

Literature Received:
Corporate Overview . . . . No
List of HIS Modules . . . . Yes
List of Services . . . . No
Complete description of each module . . . . No
Innovative Peripherals . . . . No
Third-party reviews . . . . No

4.4.2.1 Overview

Unfortunately, SMS only sent to Kensal Corporation literature describing their SMS OPENLab, a client/server laboratory information system (LIS). Since an LIS is a subset of an HIS, a brief overview of SMS and their the OPENLab system is presented.

4.4.2.2 Voice Recognition And Multimedia

SMS OpenLab supports voice recognition and multimedia technology. Examples of multimedia features include on-line Help, CD-ROM reference manuals, scanned images for user-tailored Help files, full motion video and potential links to hospital satellite connections for remote training sessions, documentaries and network-wide continuing education and training opportunities.

4.4.2.3 Encoding Enterprise Rules

OPENLab automates administrative tasks and exception alerts while eliminating redundancy. Operational and clinical rules capabilities are embedded into OPENLab. For example, users can set up results reporting based on criteria such as location, choice of print media, day of the week or time, to ensure that results are delivered to the appropriate clinicians immediately and in the format they desire.

4.4.2.4 Open Systems Approach

OPENLab is based on an open system approach, enabling users to choose the technology and operating system that best fits their needs. Users may use off-the-shelf software such as report writers, spreadsheets, databases and word processing applications. Optionally, an OPENLab system includes an HL7/ASTM compliant interface engine to optimize network and system communications. Further, full support of point-of-care testing devices, faxes, printers and pagers in physician offices is provided.

4.4.2.5 Ad-Hoc Reports

Users can define ad-hoc report formats which integrate data, text, and graphical representation of results. The need for ad-hoc reporting was underscored by SMS since the laboratory marketplace is constantly changing. Microsoft Access was cited as an example of a "canned-package" that combines the power of a relational database with an easy-to-use graphical report writer.

4.4.2.6 Augmentable On-Line Help

Context sensitive on-line help can be augmented to include standard operating procedures, scanned images, CD-ROM reference manuals, and multi-media capabilities with full motion video. SMS claims that "any number of third party packages" may be used to include text and graphics into the Help feature.

4.4.2.7 On-Line Screen Editing

Rather than contracting SMS to alter screens every time a change is needed, an on-line screen editor is available which enables a user to tailor screens to meet individual specifications, improve system flow, and user productivity. The reconfigurable features are: the prompt text, tabbing sequence between fields, and the layout of fields over one or more screens. Changes can be executed throughout the system without bring the OPENLab system down.

4.4.2.8 Flexible Human-Interface

OPENLab is GUI-based, multitasking compliant, and has user-definable security levels. In addition to support of mice, track balls, keyboards, and "hot keys"--light pens and touch-screen data entry options are available. A common user-interface model may be applied over the client/server technology; however, entity-specific (client) tailoring is allowed for improved end-user throughput.

4.4.2.9 Platform and Network Hardware

PC, IBM RISC System/6000, Digital VAX/VMS, Alpha, HP, Ethernet LAN.

4.5 MEDITECH (Medical Information Technology, Inc.)

MEDITECH
MEDITECH Circle
Westwood, Massachusetts 02090
Phone: (617) 821-3000
FAX: (617) 329-9977

Literature Received:

Corporate Overview	No
List of HIS Modules	Yes
List of Services	No
Complete description of each module	Yes
Innovative Peripherals	Yes, Handheld Computer
Third-party reviews	Yes

4.5.1 Overview

Meditech is a software and service company who develops, installs, and supports information systems for health care organizations of all sizes. Meditech emphasizes their technical innovation such as the new Handheld Point of Care Computer[xxix], and their "enterprise-wide computerized patient records." Meditech offers perpetual license agreements, periodic enhancements, ongoing education, and free system upgrades so customers can migrate to new technologies as they develop.

Meditech has 700+ installations (as of 1994) worldwide, with a majority of the customers located in the United States, Canada, and the United Kingdom. Meditech has averaged more than 80 new customers annually during the past five years.

Meditech emphasizes a flexible, integrated approach to information systems which provide patient-based information, open systems connectivity, and easy to use decision support tools necessary for today's community health care enterprises.

Clients may build information networks comprised entirely of Meditech applications or combine Meditech's modules with other vendor's products in open networks.

Meditech boasts a design principal which mandates that information systems be easy to use. One example they point to is their PCI (Patient Care Inquiry) product, used by many physicians, and can "literally be learned in five minutes."

4.5.2 Meditech's Integrated Health Care Information System (HCIS) Products

Product	No. of Customers Licensed per Product as of 5/1/95	Introduction Date
Admissions	507	1971
Anatomical Pathology	386	1980
Blood Bank	365	1981
Case Mix Mgmt	471	1984
Clinicians' On-Line Reference	23	1992
Community-Wide Scheduling	257	1995
Departmental	347	1991
Laboratory	548	1969

Medical Records	505	1972
Microbiology	518	1970
Nursing	416	1984
Order Entry	474	1990
Patient Care Inquiry	400	1988
Pharmacy	455	1975
Radiology	365	1980
Accounts Payable	404	1978
Billing/Accts Receivable	419	1977
Budgeting & Forecasting	18	1994
Cost Accounting	132	1985
Executive Support System	207	1991
Fixed Assets Accounting	266	1981
General Ledger	406	1978
Office Automation	313	1986
Materials Management	350	1980
Payroll/Personnel	374	1982
Optical Disk Archiving	109	1993
PC Workstation Software	293	1993

4.6 KEANE, Inc.

Keane, Inc.
Healthcare Services Division
290 Broadhollow Road
Melville, NY 11747
Phone: (516) 351-7000
FAX: (516) 351-7115

Literature Received:
- Corporate Overview . . . . Yes
- List of HIS Modules . . . . Yes
- List of Services . . . . No
- Complete description of each module . . . . Yes
- Innovative Peripherals . . . . No
- Third-party reviews . . . . No

4.6.1 Overview

John F. Keane founded the company in 1965 as a sole proprietorship and in 1967 incorporated the company in Massachusetts. Keane has since grown into a $350 million company with over 4,000 business and technical professionals. Headquartered in Boston, Massachusetts, Keane provides services across a network of over 40 branch offices throughout the United States and Canada.

Keane's initial corporate objectives were to assist companies in the design, development and implementation of computer systems and provide project management services to Fortune 1000 firms. Keane is now also well known for its project management methodology, Productivity Management and for the ability to complete even the most complex projects on time and within budget.

Keane's mission is to help organizations leverage their software assets and resources to achieve their business objectives. Keane strives to build long-term, mutually beneficial relationships with its client companies by effectively addressing their software development needs. Keane's success in meeting their needs has enabled the company to derive more than 90% of its annual revenue from existing clients. It has also resulted in Keane being recognized as one of the best managed small companies in the United States by publications such as Businessweek, Forbes, Financial World and Investors Business Daily.

Keane has two operating divisions: the Information Services Division (ISD) and the Healthcare Services Division (HSD). ISD provides custom applications software for corporations with large and recurring software development needs. Application software development includes systems planning, analysis, design, and maintenance. ISD also provides project management and help desk out-sourcing for clients.

Keane's Healthcare Services Division develops and supports a full line of UNIX-based "open" hospital applications including Patient Management, Financial Management, Patient Care and Clinical Systems. The Leadership Plus Series, a PC-based Long Term Care solution is Keane's offering for the long-term care market.

Headquartered in Melville, New York, the Healthcare Services Division has branch offices in Hunt Valley, Maryland, and Los Angeles, California."

4.6.2 Division Overview

In 1984, Keane made its software available as a turnkey package. This full line of modular, yet integrated, software applications solidified Keane's reputation in the marketplace. In April of 1992, Keane acquired Ferranti Healthcare Systems Corporation, a software provider for acute-care hospitals and long-term care facilities. This acquisition expanded Keane's geographical presence in the acute and rehabilitation hospital market and added approximately 300 long-term care clients with 700 facilities located across the United States. In August of 1993, Keane acquired the software and selected assets of Professional Healthcare Systems, Inc. headquarted in Los Angeles, California. This acquisition brought to Keane a prestigious client base, including large teaching hospitals and several large healthcare chains. In April of 1995, Keane acquired the Infostat division of Community Healthcare Computing, positioning Keane among the top healthcare information systems vendors in the country and increasing Keane's install base to over 230 hospitals.

Keane currently markets and supports a full line of information systems for the healthcare environment:

Threshold: a comprehensive hospital information system, uses open system computing technologies that combine RISC-based hardware, the UNIX operating system, a fourth generation programming language, and a relational database management system.

Patcom: a proven, highly rated Patient Management System designed for large teaching hospital and multi-entity facilities.

Leadership Plus: the premier financial and resident care system for long-term care facilities.

In addition to application software, Keane offers support services that include new enhancements to meet changing regulatory requirements, hot-line, and remote diagnostics. Keane continues to offer both facilities management and transition management that provide either long-term or short-term on-site system support, training and management.

4.6.3 Threshold Hospital Information System

The following from Keane document no. 08194T:

Patient Management System

Patient Accounting:	Inpatient ADT/Census Outpatient Registration Billing/Accounts Receivable/Collections
Medical Records:	Central Index Abstracting/Reporting/DRG Grouping Chart Deficiency Chart Tracking

Financial Management Systems
Accounts Payable
General Ledger/Budgeting
Payroll/Personnel
Materials Management

Clinical Systems
Order Communications
Laboratory
Pharmacy
Radiology

Executive Management Support
Managed Care
Executive Information System
Employee Scheduling
Document Imaging

Supplemental Systems
Quality Management
Infection Control
Utilization Review
Medical Staff Administration
Home Health Information

APPENDIX A
NETWORKING AND STANDARDS

Many HIS systems connect various computer systems together within the hospital and these systems branch out to terminals for end-users. Such networks in the local environment are known as Local Area Networks. However, linkages to the HIS are not limited to within the LAN. External forces are pushing the internetworking boundaries of the HIS.

It has become difficult for hospitals to stand alone. Health care reform is driving a new health care model-- a hospital today is just one stop along an entire continuum of care that can include other providers such as physician offices, home health agencies, PPOs (Preferred Provider Organizations) and HMO (Health Maintenance Organizations). Local medical centers are joining together to become regional systems who are themselves tapping into national data resources to improve decision making and compare thier performance to others nationwide.

Organizations must share caregiver information as patients move along the continuum. They must establish two-way links with national and regional data-bases to report and use ubiquitous data critical to ascertaining risk and providing cost-effective care. As a result, today's health delivery model is three-tiered, its orientation radiating outward from the local, stand-alone organization to the regional, community-based system to the national governing organization.

The following section examines some of the network technology being used to establish these local area networks (LANs), metropolitan area networks (MANs), and wide area networks (WANs).

A.1 Ethernet, A Local Area Network Technology

Ethernet[xxx] is a local area network (LAN) technology that transmits information between computers at speeds of 10 and 100 million bits per second (Mbps). A LAN is defined as a privately owned data communications system that usually covers a relatively limited territory, hence the term "local area."

Currently the most widely used version of Ethernet technology is the 10-Mbps twisted-pair variety. The 10-Mbps Ethernet varieties include the original thick coaxial system, as well as thin coaxial, twisted-pair, and fiber optic systems. The most recent Ethernet standard is the 100-Mbps system which is based on twisted-pair and fiber optic media.

A.1.1 Ethernet is a Popular, Vendor-Neutral Network Technology

There are several LAN technologies in use today, but Ethernet is by far the most popular. Networking vendors estimate that as of 1994 there were nearly 40 million Ethernet nodes installed worldwide. The widespread popularity of Ethernet ensures that there is a large market for Ethernet equipment, which helps keep the technology competitively priced.

From the time of the first Ethernet standard the specifications and the rights to build Ethernet technology have been easily available to anyone. This openness resulted in a large Ethernet market, and is another reason Ethernet is so widely implemented in the computer industry today.

The vast majority of computer vendors today provide equipment with 10-Mbps Ethernet attachments, making it possible to link all manner of computers with an Ethernet LAN. As the 100-Mbps standard becomes more widely adopted you can expect to see computers equipped with Ethernet interfaces that operate at both 10-Mbps and 100-Mbps.

The ability to link a wide range of computers using a vendor-neutral network technology is an essential feature. Most LANs today support a wide variety of computers purchased from different vendors and require a high degree of network interoperability, which Ethernet provides.

A.1.2 Development of Ethernet Standards

The specifications for Ethernet were first published in 1980 by a multi-vendor consortium that created the DEC-Intel-Xerox (DIX) standard. Ethernet technology was then adopted for standardization by the 802 LAN committee of the Institute of Electrical and Electronics Engineers (IEEE).

The IEEE standard was first published in 1985, and its formal title is "IEEE 802.3 Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications." This standard provides an "Ethernet like" system based on the original DIX Ethernet technology. All Ethernet equipment since 1985 is built according to the IEEE 802.3 standard, which is pronounced "eight oh two dot three."

To be absolutely accurate, then, we should refer to Ethernet equipment as "IEEE 802.3 CSMA/CD" technology. However, most of the world still knows it by the original name of Ethernet, and that's what we'll call it as well.

The 802.3 standard is periodically updated to include new technology. Since 1985 the standard has grown to include new media systems for 10-Mbps Ethernet (e.g. twisted-pair media), as well as the latest set of specifications for 100-Mbps Ethernet.

A.1.3 Expanding Ethernets

Ethernet was designed to be easily expandable to meet the networking needs of a given site. Individual Ethernet segments can be linked together to form a larger Ethernet LAN system using a signal amplifying and retiming device called a repeater. A given Ethernet LAN can consist of merely a single segment, or of several segments linked with repeaters. Ethernet LANs can be linked together to form extended network systems using packet switching devices.

To help expand Ethernet systems, networking vendors sell devices that provide multiple Ethernet ports. These devices are known as hubs since they provide the central portion, or hub, of a star-wired media system. There are two major kinds of hub. The first kind provides repeater ports and is known as a repeater hub. Each port of a repeater hub links individual Ethernet segments together to create a single Ethernet LAN.

The second kind of hub provides packet switching based on bridging and/or routing ports and is known as a switching hub. Each port of a switching hub links separate Ethernet LANs together to create a larger network system composed of multiple LANs.

While an individual Ethernet LAN may typically support from a few up to several dozen computers, the total system of Ethernet LANs linked with bridges or routers at a given site may support many hundreds or thousands of machines.

A.1.4 Elements Of The Ethernet System

The Ethernet system consists of three basic elements: 1. the physical medium used to carry Ethernet signals between computers, 2. a set of medium access control rules embedded in each Ethernet interface that allow multiple computers to arbitrate access to the shared Ethernet channel,

and 3. an Ethernet packet, or frame, that consists of a standardized set of bits used to carry data over the system.

Computers attached to an Ethernet send application data to one another using high-level protocol packets, such as the TCP/IP protocol used on the worldwide Internet. These high-level protocol packets are carried between computers in the data field of Ethernet frames. The system of high-level protocols carrying application data and the Ethernet system are independent entities that cooperate to deliver data between computers.

A given Ethernet system can carry several different kinds of high-level protocol data. For example, a single Ethernet can transmit data between computers using the vendor-neutral TCP/IP protocols as well as the more vendor-specific Novell or AppleTalk protocols. The Ethernet is simply a trucking system that carries packages of data between computers; it doesn't care what is inside the packages.

For more information on Ethernet, see the on-line quick reference book, by Charles Spurgeon, through the WWW URL: http://wwwhost.ots.utexas.edu/ethernet/descript-10quickref.html.

A.2 Asynchronous Transfer Mode (ATM)

In some multi-hospital networks, ATM (Asynchronous Transfer Mode) technology is being used as a basis for sharing information along the continuum of health care. ATM[xxxi] allows interoperability of information, regardless of the "end-system" or type of information. ATM is an "emerging technology" driven by international consensus, not by a single vendor' s view or strategy.

Historically, there have been separate methods used for the transmission of information among users on a Local Area Network (LAN), versus "users" on the Wide Area Network (WAN). This situation has added to the complexity of networking as user' s needs for connectivity expand from the LAN to metropolitan (MAN), national, and finally world wide connectivity. ATM is a method of communication which can be used as the basis for both LAN and WAN technologies. It is felt that over time as ATM continues to be deployed, the line between local and wide networks will blur to form a seamless network based on one standard-ATM.

Today, in most instances, separate networks are used to carry voice, data and video information-mostly because these traffic types have different characteristics. For instance, data traffic tends to be "bursty"-not needing to communicate for an extended period of time and then needing to communicate large quantities of information as fast as possible. Voice and video, on the other hand, tend to be more even in the amount of information required-but are very sensitive to when and in what order the information arrives. With ATM, separate networks will not be required. ATM is the only standards based technology which has been designed from the beginning to accommodate the simultaneous transmission of data, voice and video.

A.2.1 ATM Technology

ATM is available at various speeds from Megabits to Gigabit speeds. When information needs to be communicated, the sender *negotiates* a "requested path" with the network for a connection to the destination. When setting up this connection, the sender specifies the type, speed and other attributes of the call, which determine the end-to-end quality of service.

Another key concept is that ATM is a switched based technology. By providing connectivity through a switch (instead of a shared bus) several benefits are provided: 1) dedicatedbandwidth per connection, 2) higher aggregate bandwidth, 3) well defined connection procedures, and 4) flexible access speeds.

Using ATM, information to be sent is segmented into fixed length cell, transported to and re-assembled at the destination. The ATM cell has a fixed length of 53 bytes. Being fixed length allows the information to be transported in a predictable manner. This predictability accommodates different traffic types on the same network.

The cell is broken into two main sections, the header and the payload. The payload (48 bytes) is the portion which carries the actual information-either voice, data, or video. The Header (5 bytes) is the addressing mechanism.

A.2.2 ATM System Architecture

ATM is a layered architecture allowing multiple services like voice, data and video, to be mixed over the network. Three lower level layers have been defined to implement the features of ATM.

The Adaptation layer assures the appropriate service characteristics and divides all types of data into the 48 byte payload that will make up the ATM cell.

The ATM layer takes the data to be sent and adds the 5 byte header information that assures the cell is sent on the right connection.

The Physical layer defines the electrical characteristics and network interfaces. This layer "puts the bits on the wire." ATM is not tied to a specific type of physical transport.

A.2.3 The Status of ATM Technology

ATM has moved from concept to reality with products and services available today. The ATM Forum, discussed further in section 4.3.4, has sponsored interoperability demonstrations to prove the technology and continues to meet to discuss the evolution of ATM.

ATM coexists with current LAN/WAN Technology. ATM specifications are being written to ensure that ATM smoothly integrates numerous existing network technologies, at several levels (i.e., Frame Relay, Ethernet, TCP/IP). Equipment, services and applications are available today and are being used in live networks.

A.2.4 The ATM Forum

The ATM Forum was started in October of 1991 by a consortium of four computer and telecommunication vendors. In June 1994, the forum had over 700 members. Membership is made up of network equipment providers, semiconductor manufacturers, service providers, carriers and end users.

The Forum is not a Standards body. The ATM Forum is a consortium of companies that writes specifications to accelerate the definition of ATM technology. These specifications are then passed up to ITU-T (Formerly the CCITT) for approval. The ITU-T standard body fully recognizes the ATM Forum as a credible working group.

For more information on the ATM Forum and the ATM technology:

Email:	info@atmforum.com
WWW URL:	http://www.atmforum.com/

Appendix B: Glossary of Telemedicine and Hospital Information Systems Acronyms

Acronym	Definition and Comments
ABI	Application Binary Interface
ACH	Automatic Clearing House
ACR	American College of Radiology
ADC	, Analogue to Digital Converter
ADT	Admission Discharge Transfers
ANSI	American National Standards Institute
APG's	Ambulatory Patient Groups
API	Application Program Interface
APM	Anatomical Pathology Module, part of an HIS
AR	Accounts Receivable
ARPA	Advanced Research Projects Agency
ASCII	American Standard for Code Information Interchange
ASTM	American Society for Testing and Materials
ATA	American Telemedicine Association, 512-480-2247
ATIS	Alliance for Telecommunications Industry Solutions
ATM	Asynchronous Transfer Mode. Automatic Teller Machine, Adobe Type Manager
AUI	Attachment Unit Interface, Ethernet transeiver cable between actual interface (computer) and the MAU
B-Channel	Bearer channel, ISDN channel with 64 kbps bandwidth (see PRI)
B/AR	Billings, Accounts Payable
BAI	Basic Access Interface, ISDN with two B and one D channels (2-64kbps, 1- 16 kbps), (2B+D)
BLOB	Binary Large Object
BNC	A common type of quarter twist connector for coaxial cable.
BRI	Basic Rate Interface-16kbps ISDN Channel
CAP	College of American Pathologists, Central Arizona Project
CCD	Charge Coupled Device, uses Photovoltaically generated packets of charge that are converted to pixels.
CCITT	Standards group now called ITU-T
CDR	Clinical Data Repository
CEN	European Standards Group
CEN/TC 251/WG4	Working on spec similar to HL7. (CEN and HL7 coordinate)
CEO	Chief Executive Officer
CHCS	Composite Health Care System
CHIN	Community Health Information Network
CIS	Clinical Information System
CLK	Clerk
COO	Chief Operating Officer
COPE	Combined Patient Experience, a laboratory medicine database
CORBA	Common Object Request Broker Architecture
COSI	Corporation for Open Standards International
COTS	Connection-Oriented Transport Service
CPE	Customer Premises Equipment
CPR	Computer-based Patient Record; Coronary Pulmonary Resuscitation
CPT	Current Procedure Terminology
CPU	Central Processing Unit
CQI	Continuous Quality Improvement
CR	Computed Radiography
CSMA/CD	Carrier Sense Multiple Access with Collision Detection, Ethernet features
CT	Computed Tomography
D-channel	Delta-channel, ISDN channel with 16 kbps bandwidth (see BRI)
DDN	Defense Data Network
DEC	Digital Equipment Corporation
DG	Data General Corporation
DHCP	Decentralized Hospital Computer Program, used by DEC and CHCS
DICOM	Digital Imaging and Communications in Medicine
DINS	Digital Imaging Network Systems - Military term

Appendix B: Glossary of Telemedicine and Hospital Information Systems Acronyms

DIX	DEC Intel Xerox, initial standard for Ethernet (now an IEEE 802.3 standard)
DMSSC	Defense Medical Systems Support Center
DNA	Deoxyribonucleic Acid Sp?
DoD	Department of Defense
DoH	Department of Health
DRAM	Dynamic Random Access Memory
DRG's	Diagnosis Related Groups
DSOs	Digital voice channels, used with ISDN
DTE	Data Terminal Equipment, usually a computer that interfaces with Ethernet
DTS	Dietetics
DVA	Department of Veterans Affairs
Dx	Diagnosis
EDI	Electronic Data Interchange
EDIFACT	Electronic Data Interchange for Administration, Commerce, and Transport
ENR	Enterprise Network Roundtable, user group of ATM
EOC	Expense Operating Center ? ,an accounting term
EOQ	Economic Order Quantity
EPI	Enterprise Patient Index
EPROM	Electronically Programmable Read Only Memory
ESS	Executive Support System
FAQ	Frequently Asked Questions
FCS	Full Cover Slip
FDDI	Fiber Distributed Data Interface
FIFO	First In - First Out
FOE	Fiber Optic Enclosure
FOIRL	Fiber Optic Inter-Repeater Link, used with Ethernet
FOMAU	Fiber Optic Medium Attachment Unit, transceiver for Ethernet
FTP	File Transfer Protocol
FYI	For Your Information
GAO	General Accounting Office
GATT	General Agreement for Tariff and Trade
GHNet	Global Health Net
GL	General Ledger
GNA	Global Network Academy
GNP	Gross National Product
Gopher	animated contraction of "go-for" looks for subject or words of interest on the NET
GP	General Practitioner
GPIP	General Purpose Image Processing
GPR	Graphical Patient Record
GRIPE	Group for Research in Pathology Education
GUI	Graphical User Interface
GYN	Gynecological
HAF	Hyperalimentation Fluids
HCIS	Health Care Information System
HCO	Health Care Organization
HCTG	Health Care Technology Group
HDTV	High Definition Television
HIS	Hospital Information System, Health Information System
HISPP	Health Informatics Standards Planning Panel
HISPP	Health Informatics Standards Planning Panel, formed by ANSI
HISS	Hospital Information Support System
HITS	Health Innovations in Technology Systems, yearly award given by the Henry Ford Health System
HL7	Health Level 7
HMO	Health Maintenance Group
HTML	Hyper Text Markup Language

Appendix B: Glossary of Telemedicine and Hospital Information Systems Acronyms

IBM	International Business Machines
ICD	billing code used for various cases?
ICU	Intensive Care Unit
ID	Individual Identifier
IDN	Integrated Digital Network
IEC	Image Exchange Committee, developing Pathology extension to DICOM
IEEE	Institute of Electrical and Electronic Engineers
IHID	Inter Hospital Image Distribution
IPA	Independent Physicians Association, or Independent Practice Association
ISA	International Standards Association, Instrumentation Society of America
ISAM	Indexed Sequential Access Method, (used with data bases)
ISDN	Integrated Services Digital Network
ISIS	Information System-Imaging System
ISO	International Standards Organization
IT	Information Technology
ITU-T	International Telecommunications Union-Telecommunications, sets ISDN standards
JPEG	Joint Photographers Expert Group
JWG-CDM	Joint Working Group, Common Data Model
LAB	Laboratory
LAM	Lymphangioleiomyomatosis
LAN	Local Area Network
LANL	Los Alamos National Laboratory(ies)
LEOS	Low Earth Satellite
LIFO	Last In - First Out
LIS	Laboratory Information System
LM	Laboratory Module, part of an HIS
LOS	Length Of Stay
LSDA	Line Scan Diode Array, provides high resolution large image scanning capability
MAC	Medium Access Control, provides access when available from each Ethernet station
MAR	Medication Administration Record
MAU	Medium Attachment Unit, Transceiver for Ethernet that interfaces between computer and the medium.
MD	Medical Doctor
MDC	MUMPS Development Committee
MDF	MD Forms
MDI	Medium Dependent Interface, Ethernet hardware that connects directly interfaces to the medium.
MDIS	Medical Diagnostic Imaging Support system - used by Military.
MDL	MD Lookup
MDR	MD Retrieval
MIS	Medical Information Systems, Management Information System
MPI	Master Patient Index
MRI	Magnetic Resonance Imaging
MRN	Medical Record Number
MSDS	Message Standards Developers Subcommittee, health care message interchange stds, formed by HISPP
MTF	Medical Treatment Facilities - Military Term
MUMPS	Massachusetts (Gen. Hosp.) Utility Multi Programming System, Prog. Lang. used by SAIC & some Hosp.
NCPDP	National Council of Prescription Drug Pharmacies
NEMA	National Electrical Manufacturers Association
NET	Short for Internet
NHS	National Health Services?
NII	National Information Infrastructure, goal to provide equable information services to all Americans
NII-HIN	National Information Infrastructure-Health Information Network
NIST	National Institute of Standards and Technology
NMF	Network Management Forum
NOS	Not Otherwise Specified
NPS	Non Printed Specifics

NRS	Nursing
ODA	Optical Disk Archiving system
ODJ	Optical Disk Jukebox. Optical media (platters) for high density digital storage
OLE	Object Linking and Embedding
OMG	Object Management Group. (responsible for CORBA standards)
OSI	Open System Interconnection. seven layers of of hierarchy
OT&E	Operational Test and Evaluation
PACS	Picture Archiving and Communications System - Used by Military
PAD	Patient Administration Department?
PAS	Patient Appointment Scheduling
PBXs	Private Branch Exchanges
PC	Personal Computer
PCI	Patient Care Inquiry. high speed buss that carries information in PC's and Power Macs
PCM	Personal Computer-Microscope. provides workstation features with digital images
PET	Positron Emission Tomography
PHO	Physician Hospital Organization
PHR	Pharmacy
PMED	Portable Medical Entry Device
PO	Purchase Order
POE	PowerOpen Environment
PPO	Preferred Provider Organizations
PRI	Primary Rate Interface-ISDN 23 ea. 64 kbps channels + one 64-kbps D-channel
PRO	Peer Review Organization
PtCT	Patient Care Technologies, Inc.
QA	Quality Assurance
QC	Quality Control
QTD	Quarter To Date
R&D	Research and Development
RAD	Radiology
RAID	Redundant Array of Inexpensive Disks
RBOCs	Regional Bell Operating Companies. the 7 Baby Bells
RBRVS	
RGB	Red Green Blue. a TV full color generating scheme where all color is obtained by addition of R.G.B
RIS	Radiology Information System
RM	Reference Model. Radiology Module. part of an HIS
RN, R.N.	Registered Nurse
RNA	Ribonucleic Acid
RTE	Remote Terminal Emulation
SAIC	Science Applications International Company
SCSI	Small Computer Standard Interface. pronounced "scuzzi"
SDC	Surgical Day Care
SDOs	Standards Developing Organizations
SMTP	? , information protocall
SNMP	
SNOMED	Systematized Nomenclature of Medicine
SOW	Statement of Work
SQL	Structured Quiry Language
SRDRG's	Severity Refined Diagnosis Related Groups
SSN	Social Security Number
T1	Communication lines with 1.54Mbt/sec transmission rate
TA	Terminal Adapter. interfaces with ISDN
TAMC	Tripler Army Medical Center
TCP/IP	Transfer Control Protocol. Internet Protocol
TDS	? , Total Dissolved Solids
TE	Terminal Equipment. devices using ISDN to transfer information

TELNET	Information Protocall
TM	Telemedicine
TQM	Total Quality Management
TRP	Technology Reinvestment Project
UR	Utilization Review
URL	Universal Resource Locator
VA	Veterans Administration
VAR	Value Added Reseller
VRAM	Video Random Access Memory
WAN	Wide Area Network
WHIN	Wisconsin Health Information Network
WOM	Write Only Memory. Useful for storing your mother-inlaw's address
WORM	Write Once Read Many - Type of memory
WSU	Work Storage Unit, usually very high density digital storage may have fiber optic data transmission.
WTO	World Trade Organization
WWW	World Wide Web, graphical interface with hypertext used on the NET
XIWT	Cross-Industry Working Team, working on framework for the National Information Infrastructure
YTD	Year To Date

5. PATHOLOGY IMAGES AND INFORMATION SYSTEMS

This section consists of information on various anatomic pathology software and on a search for pathology images on the Internet.

5.1 Anatomic Pathology Modules

Information on anatomic pathology software modules was collected from telephone interviews with individuals at the various companies that provide them

5.1.1 Phil Mullarky, Sunquest Information Systems Inc., June 12, 1995

The module produced by Sunquest is called the *FlexiLab Anatomic Pathology/Cytology System*. It is a stand alone system in which the HIS system is built "from scratch" with its own pathology module. There are a total of 250 installation sites.

5.1.1.1 Imaging

The *FlexiLab Anatomic Pathology/Cytology System* is not capable of handling images as of right now, but the company is planning on it in the future. Mr. Mullarky stated that the images will be digital and will be in color. He said that you "must have color if you are working with pathology." The images stored under this system will be referenced to/coincide with cases from patients.

According to Mr. Mullarky, the storing capacity of the system is unlimited, and will be determined by the type of disk used.

Images will be able to be shared by pathologists. When asked specifically how the inquiries will be made, Mr. Mullarky just said the images will be referenced by a case. A local area network will be used, and no data rate (pixels/sec. or images/sec.) has been determined yet.

5.1.1.2 Voice

Sunquest is currently in the process of installing voice recognition with Kerzweil.

5.1.1.3 User Interface

The FlexiLab Anatomic module utilizes a mouse and a keyboard.

5.1.1.4 Host Computers

Sunquest supports UNIX, VM, and VMS platforms. As for the types of microprocessors, Mr. Mullarky said that Digital and IBM were used.

5.1.1.5 Cooperation

Sunquest is not working with any professional societies and/or government agencies while developing their system. They are, however, inspected by the FDA.

As far as working with us as we develop our workstation for pathology, Mr. Mullarky said that that was "to be determined." He stated that we might be on the same path as their company, and could not answer the question at the moment. He will be sending some literature on their system for us to review.

5.1.2 Nancy Vetter, Marketing Manager of Dynacor Inc., June 13, 1995

The pathology module produced by Dynacor Inc. is called the *PREMIER Series*. It can be a stand alone system, but Dynacor usually sells it with a clinical laboratory information system . It can also interface with hospital information systems. There are a total of eight installation sites.

5.1.2.1 Imaging

The system Dynacor uses is the IBM AS/400 System. It has the capability to handle images, but Ms. Vetter stated that no one wants to use images yet, so they have not implemented it into their system. The images would be digital and would be in color, Ms. Vetter assumed. As far as the images being used for reference or for patients files, Ms. Vetter said that at the design level, there was the capability for both.

Mr. Vetter did not know any information about the potential storing capabilities of the system. She said those questions would have to be directed toward the Development Manager.

The images will be shared between pathologists upon authorization. Ms. Vetter said that she assumed pathologists would want to have this built into their system. The images would be shared by a file server, and would be over either a local are network or wide are network. The type of network used would depend on the institution.

5.1.2.2 Voice

The *PREMIER Series* does not have voice yet, but does have the capability of implementing it. Ms. Vetter stated that voice was not a priority and therefore not utilized. If voice is installed, it will most likely be voice recognition and synthesis.

5.1.2.3 User Interface

Currently the module utilizes only a keyboard, but has "hot spots" for the mouse.

5.1.2.4 Host Computers

Dynacor supports IBM type computers (the AS/400). Ms. Vetter said that if someone wanted microprocessors, IBM compatibles would be used.

5.1.2.5 Cooperation

Dynacor does not work with any professional societies and/or government agencies in the development of their system. They work with user space (with their own organization). Occasionally they work with a new client or focus group (two or more clients).

Ms. Vetter stated that they would very much like to work with us in the development of our workstation for pathology. In addition, she has sent some literature about their company and the anatomic pathology module for us to review.

5.1.3 Mary Wehlacz, Citation Computer Systems Inc., 6-14-95

The pathology module produced by Citation Computer Systems Inc. is called *Citation APS*. It is developed by building a laboratory information system (LIS) with an AP module that can be added on. It is not a stand alone system. There are 20 installation sites.

5.1.3.1 Imaging

Citation APS does not handle images yet, however, the company is actively looking into it. Ms. Wehlacz was not certain if the images would be digital or analog, but said that they would be in color and had the possibility of being either reference images or new images for patients files.

Ms. Wehlacz did not know what the storing capacity of the module would be for images.

Images will be shared between pathologists with a file server under *Citation APS*. The images will be on a network that PC's will be able to access when attached.

5.1.3.2 Voice

The company's system does not have voice. They are looking into voice recognition, but Ms. Wehlacz stated that their systems are not ready for it yet. She said they have tested voice but the system has problems recognizing certain words. When it does not recognize a word, it will form a list of words to choose from.

5.1.3.3 User Interface

Currently *Citation APS* utilizes a touch screen, mouse, and keyboard for its system.

5.1.3.4 Host Computers

The platforms supported by Citation APS are a local area PC based client servers. If the company were to use microprocessors, they would use Intel and have a minimum of 46 processors.

5.1.3.5 Cooperation

Citation Computer Services Inc. works with customers from a variety of hospitals (teaching, etc.) in the development of their pathology module.

Ms. Wehlacz stated that they would be very interested in working with us in the development of our pathology workstation. She is sending literature on their system to Kensal for us to review.

5.1.4 Mark Hughes, Cerner Corporation, 6-21-95

The name of Cerner Corp.'s pathology module is *PathNet Anatomic Pathology*. *PathNet* has two different architectural features of the module. The first is the traditional mainframe (stand alone) system. Mr. Hughes explained that the hardware for this system is too expensive to actively market imaging with, and is not cost effective for clients. Only two clients currently use the system as a stand alone system. Clients must purchase both the PathNet Core and the PathNet Pathology System in order to use it as such. Mr. Hughes added that although the clients do like the stand alone system, very few utilize it. The second architectural feature of the module is a client server system where PC's can be purchased and used. This type of system is not a stand alone system. Cerner Corp. is currently in a transitional phase of switching over to the client server architecture. This system will use Microsoft Windows, which makes images easier to manage, along with the possibility of using real-time. This client server architecture will cost less because the hardware will cost less. Currently, there are 420 installation sites of the traditional mainframe system.

5.1.4.1 Imaging

Cerner Corp. has experimented in imaging with *PathNet Anatomic Pathology*. They worked in conjunction with Sony and Baylor University in image capturing with a system they referred to as

CPP, Cerner Pathology P.A.C.S (picture archiving system). With a camera mounted on top of a microscope, they would capture a microscopic image, save the image onto a disk, and then transfer it to a remote site within the hospital. A monitor was placed in a surgical suite and used as a communication vehicle between the surgeon and a pathologist. The system worked very well and the doctors liked what they saw, Mr. Hughes said. The diagnostic quality of the images were also approved by the doctors who used the system.

The images in this experiment were digital and in color (Mr. Hughes knew that it was greater than 256, but could not give an exact number). Mr. Hughes did not know the size of the images (in pixels) or the number of bits per pixel. In future use, the images on the system could be used either for reference or as images for patient files.

The storing capability of the pathology module will depend on the medium device. Currently the company uses super high density disks from Sony, which store 20 megabytes a piece.

The method of indexing employed may be a number of things. Images may be indexed by patient number, social security number, or patient name, for example. In addition, Mr. Hughes added that the SNOMED coding may be used.

Retrieval of the experimental images at Baylor took an average of 20-30 seconds. Those images captured will be stored permanently (for legal reasons of the experiment). Mr. Hughes speculated that images captured in the future will be stored for as long as the storage device will allow.

Cerner Corp.'s module is not set up for networking together, so images may not be shared between pathologists in that sense. Mr. Hughes stated that the way sharing works currently is that pathologists walk up to one station which stores the images and pulls them up from that station. There is no networking of the images though. If networking is implemented, Mr. Hughes said that both a local area network and a wide area network could be used. He did not know what the data rate would be.

5.1.4.2 Voice

Cerner Corp. has not developed voice for their system yet, but it does have the capability to be used. If they do use voice, it will be voice recognition. The company did do an interface to Covington Hospital in Virginia using the Kerzweil System (voice recognition). Problems were encountered, however, when the system did not recognize a word. When this happened, it would pull up a list of words for the pathologist to choose from.

5.1.4.3 User Interface

PathNet Anatomic Pathology utilizes a mouse and a keyboard.

5.1.4.4 Host Computers

The platforms supported by the module are the IBM Risk/6000 Box and DEC Alpha (along with other Deck products). The company does not use microprocessors at the moment, but will use Intel for future products.

5.1.4.5 Cooperation

Cerner Corp. invites clients (experts) to review and test their systems. These experts are usually pathologists, histotechs, or cytotechs from larger universities.

Mr. Hughes showed great interest in working with us as we develop our pathology workstation. He stated that he thought it would be a great option to explore. He is going to send some literature on their CPP system for us to review.

5.1.5 Carol Donohue, Director of Marketing, CoMed, July 2, 1995

The name of CoMed's anatomic pathology module is *CoPath.* It is a stand alone system. In regards to the March 1995 *CAP Today* article, *CoPath* has 186 installed sites, three of which are in Arizona (Mayo Clinic in Scotsdale; Scotsdale Memorial; and John C. Lincoln in Phoenix).

5.1.5.1 Imaging

Currently CoPath can handle "very minimal hand drawn images, such as organ diagrams." Ms. Donohue is not sure if the images are digital or analog, but stated that they are scanned with an HP Scanjet. She assumes they are digital. The size of the images is very limited because they must fit into a certain space on the patient reports. The exact size is not known. Since the images are used mainly for standard reports that are passed from doctor to doctor, Ms. Donohue said that they are strictly in black and white. The doctors did not want color.

The images are stored in "chunks," a months database at a time. They are stored in a textbox and then uploaded into the months database. Images are usually associated with a specimen number. The file is structured in a hierarchical structure. Ms. Donohue was unsure if the images could be retrieved for viewing purposes, unless they were looked at under WordPerfect. Images are stored permanently as of right now.

Ms. Donohue mentioned that they can interface to external systems which can handle other, larger images.

CoPath is a multi-user system, in which images can be shared through a file server with WordPerfect. Otherwise, a lat protocol is used to get at information. A 232 connection is used for this. The system can run with other applications, but it is not a true network. She did not know the data rate of the system.

5.1.5.2 Voice

The CoPath system does not have voice, but interfaces to other systems which usually have voice recognition. If the other system also has synthesis, then that is implemented also.

5.1.5.3 User Interface

The pathology module utilizes a touch screen and a keyboard.

5.1.5.4 Host Computers

The platforms supported by *CoPath* are PC, IBM RISC, and VAX mini-systems. All microprocessors are Intel.

5.1.5.5 Cooperation

Ms. Donohue stated that they do not work directly with any professional societies or government agencies in developing their system. They do meet regulations, however, and work with the College of American Pathologists (CAP) occassionally.

Ms. Donohue was very hesitant in giving any sort of commitment to working with us as we develop our workstation. She was concerned mainly about competition. She is sending literature on their system for us to have on file.

5.1.6 John Edmondson, VP Sales of Community Health Computing Inc., July 7, 1995

(Mr. Edmondson is forwarding the information to Fred Tillman who is the Development Manager of CHC.)

Community Health Computing Inc. (CHC) has two pathology systems. Their oldest system is about 20 years old and is called LabCare. It is a stand alone system and is what a majority of their clients use. They are in the process of building a newer system, however, called LabStat. It too will be a stand alone system, and 70% of it is already engineered. It is currently in one beta site. The questions asked in the interview were answered in regards to the newer module. There are a total of 91 installations for the older system.

5.1.6.1 Imaging

The LabStat module will be able to handle images when it is completed. The images can be either digital or analog, but Mr. Edmondson said they will probably be primarily digital. They will also be produced in color. Mr. Edmondson explained that the way their system will work is that the images will be created somewhere else and then transferred to their system. Therefore, the images will probably be used for reference. As far as the size of the images, that is unknown. The images will be used for reporting clinical information and associated with a document. Mr. Edmondson stated that they will not be used for diagnosis, just for informational purposes.

Mr. Edmondson did not know the answers to the questions regarding storing capacity and suggested that they be asked to Mr. Tillman.

The images can be shared between pathologists if they allow for it (a matter of security). Mr. Edmondson was not sure how those images would be shared though. Their system will use both a local area and a wide area network. Mr. Edmondson explained that they must build in accordance to how things are on the information highway, and that is why both networks will be used. He did not know what the data rate of the network would be.

5.1.6.2 Voice

LabStat has no application for voice today because of limitations encountered (such as it can't do conversations or recognize certain words). Mr. Edmondson stated that cytology would probably benefit more from voice because they use just one word or number to often describe a case (much easier to understand).

5.1.6.3 User Interface

CHC's module will utilize a mouse and a keyboard. The keyboard will be either a terminal or character driven interface for Windows.

5.1.6.4 Host Computers

The platform that is currently supported in the single beta site of the new system is Hewlett-Packard. They will move onto others in about a year or so. Microprocessors are from Intel.

5.1.6.5 Cooperation

CHC does not market to federally funded agencies because they did not like the older system. Other than that, they do work with clients and consultants in the development of their system.

There are no installations of LabStat in Arizona, but there are a couple of the older system, LabCare. The ones Mr. Edmondson could remember were Sun Health Boswell Hospital in Sun City and St. Joseph's in Phoenix.

Mr. Edmondson said that they would be interested in working with us as we develop our workstation. He stated that they would like to capture what we do, and that they are planning to install their system in a large, highly recognized institution in 1996 (not through the DoD). He also explained that they will be starting to build their software in six to nine months, so now would be a good time to interface and work together.

Mr. Edmondosn is sending literature to us regarding their systems.

5.1.7 Dan Callaher of MEDITECH (Medical Information Technology Inc.), July 7, 1995

(I spoke with Dan Callaher who is one of the Marketing personnel, but he was going to pass the letter and article onto Larry Gay who is another Marketing personnel who handles companies in the Southwest US.)

The name of MEDITECH's pathology module is *Meditech Anatomic Pathology*. It can be built as either a stand alone system or to be integrated into HIS systems. There are 26 applications with which it can be integrated, and 364 installation sites.

5.1.7.1 Images

Mr. Callaher stated that *Meditech Anatomic Pathology* module can handle images, but does not right now because of limited technology. He said that when the proper resolution, monitors, and so on are figured out on the technological end and become affordable, then the systems will be able to handle images. As far as whether the images will be digital or analog, Mr. Callaher said that which ever produces the best quality or resolution is what they will use. He applied the same statement to the question of color or black and white images. Depending on the cost, images will be associated with patient files, but will also be able to be used for reference.

As far as storing capability, Mr. Callaher said that there would be no limit on how many images the module would be able to handle. He was not sure about the method of indexing, but guessed it might be in a sequence query language. Since their system is hierarchical right now, he guessed that that was how the file would also be structured. The retrieval of images and the amount of time the images are saved for will depend on the technology.

Mr. Callaher stated that images could be shared between pathologists. They would be shared on a "network of some sort," he said. When asked if they would use a local area network or a wide area network, Mr. Callaher said that it did not matter. He did not know what the data rate would be.

5.1.7.2 Voice

MEDITECH's pathology module does not have voice, but they do interface to the Kerzweil product which uses voice recognition.

5.1.7.3 User Interface

The *Meditech Anatomic Pathology* module primarily utilizes a keyboard, but can also use a mouse.

5.1.7.4 Host Computers

Currently MEDITECH supports DEC and Data General platforms, but they are moving to a new system of Windows, the NT Operating System. The microprocessors they use are Intel.

5.1.7.5 Cooperation

MEDITECH works with "everything and everyone" as far as professional societies and government agencies. They have over 800 installations throughout the US., and have several in Arizona. Mr. Callaher knew that their system was implemented in the St. Mary's and St. Joseph's Hospitals (Carondolet), but was not sure of the others. He said to check with Mr. Gay for further information where that was regarded.

As far as working with us in the development of our pathology workstation, Mr. Callaher said that we must check with the joint systems people.

They do have literature available to send to us. Mr. Callaher again said that Larry Gay will call regarding that.

5.1.7.6 Telephone Interview with Larry Gay, 8-31-95

While interviewing Mr. Gay, he reiterated a lot about the system which Mr. Callaher had already stated. Their system does not handle images at the moment, but they are planning on having them in the future. The types of images they will carry will be any kind of digitized image. Images will be in color he assumed. The storage capability of the module will not be limited, he guessed, and will depend on the hardware from the third party vendor. The type of indexing they currently employ for their reports, not images, is proprietary. Mr. Gay estimated that the retrieval time of images will be much like the retrieval time of the reports - "seconds." The company uses optical storage, and therefore stores everything permanently.

Images will be able to be shared between pathologists. They will be done so through an optical or jukebox file server. The type of network they currently use is the TCPIP mode. In addition, both a wide area and local area network will be used for images. Mr. Gay did not know the data rate off hand.

Again, the system does do an interface to Kerzweil for voice recognition. The system also primarily utilizes a keyboard. The platforms supported are DEC and Data General, while the microprocessors are Intel. Mr. Gay said that the amount of memory on the hard drive varies depending on the institution supplying it.

Mr. Gay also confirmed that they do have installations in Arizona. Though he would not give specific names of hospitals, he said they do have a corporation set up with Carondolet.

Literature is being sent for us to review.

5.1.8 Steve Tablak, Marketing Manager of Tam Support Services, July 11, 1995

Tam Support Services is one of the smaller companies contacted in this interview, with 34 installations of their module. The name of their module is *PowerPath Anatomic Pathology System* and it is a stand alone system.

5.1.8.1 Imaging

As far as handling images, the *PowerPath Anatomic Pathology System* can only incorporate them into a final report. The images are digital, and the size of them depends on whatever the customers want to do. Mr. Tablak explained that most of the images are gross images or diagrams, and therefore the size is usually small. He said that most images are not viewed by doctors for diagnostic reasons, and only one or two customers use them for reference. He also stated that it does not matter if the images are in color or in black and white, as long as they are supported by Windows.

The storing capability of the module is part of the word document (Mr. Tablak did not say how many images their system could store based on that information). Mr. Tablak stated that the method of indexing employed does not matter, but they typically use accession numbers. In addition, files are structured by relations. The retrieval time of images depends on the computer and if the images are compressed, but Mr. Tablak stated that this feature is not used a lot. Since images are used for final reports, they are stored permanently.

The images can be shared between pathologists, and are done so with the Novell networking system. A local area network is used, and the data rate is 10 megabits per second.

5.1.8.2 Voice

PowerPath Anatomic Pathology System interfaces to Kerzweil to implement voice recognition on the system.

5.1.8.3 User Interface

Tam Support Services module utilizes a touch screen, mouse, and keyboard, with primary emphasis on the mouse and keyboard (Windows based).

5.1.8.4 Host Computers

Mr. Tablak stated that they can support any PC type platform. More specifically, he mentioned Novell, NT, DEC, and HP. The microprocessors they use are Intel or DEC alpha.

5.1.8.5 Cooperation

Tam Support Services works with teaching institutions in the development of their system. Mr. Tablak mentioned that they typically work with highly recognized institutions, such as Stanford. They do not currently have any installations in Arizona.

As far as working with us in the development of our workstation, Mr. Tablak said that they would like to talk to us about it. More importantly, he said it depends on if the teaching institutions they are working with like what we are doing.

Mr. Tablak is having his secretary send some information to us about their system.

5.1.9 Dennis Hart, Computer Trust Corporation, July 12, 1995

The name of Computer Trust Corporation's pathology module is *SURGE*. *SURGE* is a stand alone system, with the ability to interface to other HIS systems. There are 34 installations of this module.

5.1.9.1 Imaging

SURGE cannot handle images and the company is not planning on adding images to it.

5.1.9.2 Voice

The system does not have voice, but has the capability to obtain it. Computer Trust Corp. currently has a handshake deal with Kerzweil, which uses voice recognition. The reason they have not implemented voice is that it is too expensive, Mr. Hart said.

5.1.9.3 User Interface

The *SURGE* module utilizes only a keyboard.

5.1.9.4 Host Computer

The platforms supported by the module are the Novell System and UNIX. The microprocessors are all Intel.

5.1.9.5 Cooperation

Computer Trust Corporation does not work with any professional societies or government agencies in the development of their system. The man who built this system and the company, Dr. David Liberman, has a lot of experience in the necessary fields, and oversees everything that goes on. Occasionally they do refer to some consultants.

There are no *SURGE* systems anywhere in Arizona.

Mr. Hart feels that they do not cater to the same people that we work with, and therefore do not have resources that could really help us out. He did state that they might be willing to possibly help in our effort and codevelop for a specific company if requested to do so. He is sending literature on their module.

5.1.10 Dan Hellman, Technical Director, Anatrol Pathology Computer Systems, Aug. 23, 1995

Mr. Hellman is the technical director of the Anatrol Pathology Module. It is capable of being either a stand alone system, or interfacing to already existing software for HIS systems. If interfacing to an already existing system, Anatrol is capable of interfacing to whatever HIS system the hospital has (no preferences).

5.1.10.1 Imaging

Anatrol is not yet capable of handling images, but Mr. Hellman said that they are definately planning on it. They will carry any type of image the user would like (gross, microscopic, X-ray) as long as the file is in the proper industry format (GIF, JPEG, etc.). More than likely, the images will also be digital. Mr. Hellman explained that everything is being put on a binary or "blob." As far as color or black and white images, it will be whatever the client prefers. The software that the system has to handle the images will be canned. Mr. Hellman explained that he likes to attach to whatever already exists.

The storing capability of the module will be limited by the hardware. Text headers will be used as the method of indexing, and more than likely the images will be coded for by the SNOMED system. This is how the file will be structured also. As far as the retrieval time for the images and the length of time they will be saved for, that will all depend on the functions of the hardware.

The sharing of the images will have to be determined by the particular department who employs their system, Mr. Hellman said. There will be a data lock and a confidentiality lock on their system, so it's up to the department to determine if the images will be shared or not. If they are to be shared, a data base server will be used that utilizes all kinds of networks. In addition, both local area networks and wide area networks can be used. Mr. Hellman explained that they like to try to remain as independent from hardware as possible so as to accomidate a preferred type of set-up when the time comes. The data rate will also depend on the file being used.

5.1.10.2 Voice

As of right now, Anatrol is working on voice in conjunction with Kerzweil. They are, however, working more intently on the use of hand and pen based computers.

5.1.10.3 User Interface

The main type of interface the module utilizes is the pen right now. They are, as mentioned above, working on voice and do use the keyboard, mouse, and touchscreen.

5.1.10.4 Host Computers

Because they remain independent of hardware, Anatrol does not really support any specific platforms. They do use UNIX and NT, but only on a text basis. As far as microprocessors, Mr. Hellman said that about 85% of their clients use Intel, with a couple others using Risk.

5.1.10.5 Cooperation

Mr. Hellman said that they do occassionally work with professional societies in the development of their system, but not in a big way. They mainly work with CAP, but that is from a licensing point of view.

Anatrol does not have any installations in Arizona currently. Some states that do use their system are Texas, Oklahoma, and California.

Mr. Hellman said that they would definately be interested in working with us in the future. He would like to be kept abreast of the progress we are making and everything else that is going on. He is sending literature on their older system to us to lookover and keep on file.

5.1.11 Terry Johnson, Accupath, Aug. 24, 1995

The *ACCUPATH Anatomic Pathology System* is a stand alone system as of right now. It does have the ability to interface to ORACLE and is also motim interfacible. Mr. Johnson explained that there is just no demand for interfacing at the moment. This product is ORACLE based and is on a UNIX box.

5.1.11.1 Imaging

The pathology module does not handle images, but does carry such things as graphs and charts. Mr. Johnson said that they might consider carrying images if they received a large hospital account. If that does occur, the types of images they would carry would be cytology, Pap, and microscopic, which he feels is the main interest right now. The images would be in color and would most likely be digital.

If and when the company decides to carry images, they would not keep the images in the ORACLE database. Mr. Johnson explained that they would use a pointer to an external reference. Therefore, he did not know what the storing capability of the module would be.

Mr. Johnson also said that the images might possibly be shared between pathologists. He felt that it had commercial applications. He also felt that a wide area network would have to be used.

5.1.11.2 Voice

The *ACCUPATH Anatomic Pathology System* does not have voice. Mr. Johnson said that there is no call for it and he does not really see a reason for having it.

5.1.11.3 User Interface

ACCUPATH's system utilizes just a keyboard.

5.1.11.4 Host Computer

The platforms that ACCUPATH supports are UNIX and ORACLE. They are a licensed ORACLE resaler.

When asked about the amount of memory on the hard drive, Mr. Johnson explained that they use 386's.

5.1.11.5 Cooperation

ACCUPATH does not really work with any professional societies or government agencies in the development of their system. Since they are the main users of their product, there is no need for any outside advice yet. Mr. Johnson said that they do sell there system to some small hospitals, but they do not aid in the development of the system.

ACCUPATH does not have any installations in Arizona. They do have a beta site in Fresno, CA at Hadden Laboratories. There they work with Dr. David Hadden who does anywhere from 50-60,000 Pap Smears a year and approximately 5000 surgical slides a year.

Mr. Johnson said that they would be interested in working with us if they could find a way to cover the costs. His primary interest seemed to be with Pap Smears. Mr. Johnson also said he was puzzled by how we could possible interface our two systems.

5.1.12 Denise Smith, Marketing Manager, Advanced Laboratory Systems, Aug. 24, 1995

The name of this pathology system is *PATHLAB Anatomic Pathology*. It is both a stand alone system and an interfacible system. When interfacing to HIS systems, the one most commonly used one is the HL7 format.

5.1.12.1 Imaging

The *PATHLAB* system does not handle images currently. Ms. Smith said that they could add the feature to their system, but will not do so in the near future.

5.1.12.2 Voice

The system also does not have voice. Ms. Smith said that they did some testing with it but didn't like it very much.

5.1.12.3 User Interface

As of right now, their system utilizes only a keyboard. They are planning on adding the mouse by next year.

5.1.12.4 Host Computers

The type of platform that the *PATHLAB* system supports is UNIX. The microprocessors they use are Intel.

Ms. Smith did not know off hand how much memory was on the hard drive or RAM.

5.1.12.5 Cooperation

Ms. Smith said that they worked with only some hospitals and pathologists in the development of their system.

She did not know if they had any installations in Arizona other that St. Luke's up in Phoenix. She thought that that system was just the AP system too.

Ms. Smith said that they were not really interested in working with us in the future. She explained that their main priority for next year is a client server situation. She also explained that pathologists have not been asking for imaging, therefore they are not worrying about it right now.

5.1.13 Mary Ann Lafayette, Cytology and Pathology Services Inc., Aug. 24, 1995

The Cytology and Pathology Services Software is a stand alone system. The company only has three installation sites at the moment.

5.1.13.1 Imaging

The system does not carry images, but Ms. Lafayette said the issue has been discussed. She also explained that pathologists have not been pushing for it.

There was no information on what kind of images they might possibly carry (such as digital, color, reference, etc.). There was also no information on the storing capability of their system because the hardware is purchased by the user.

5.1.13.2 Voice

The system does not have voice yet, but they are working on it. The doctor in charge of the company is currently re-engineering a voice system by the name of *Lantastic*.

5.1.13.3 User Interface

Currently the pathology module only uses a keyboard. Mr. Lafayette explained that this is still the fastest for transcriptionists. She also explained that other areas of the system that use Windows utilize the mouse, but that is a very small number and only comes about upon request.

5.1.13.4 Host Computers

The Cytology and Pathology Services Software is strictly DOS based or PC based. It operates in any network (such as Novell). They do not produce the hardware. That is up to the purchaser.

5.1.13.5 Cooperation

The company has specifically developed their system for two hospitals in Alabama to use. Those are the only people they have worked with in its development. Ms. Lafayette was also just informed that some University of Alabama cytotechs will be coming in soon to look at her system and possibly work with it. There are no installations in Arizona.

When asked if they might be interested in working with us in the development of our system, Ms. Lafayette said that she would pass the information on to the pathologist in charge to see what he thinks. A couple of days after this interview was conducted, Dr. Donald(?) Canley, the pathologist in charge, did contact me with some constructive criticism about what we are doing. He said that he was intrigued by what we were doing but also said he didn't think it would work. His main reason for this was that we would have to improve the resolution greatly in order for it to become effective. In order to do this, he suggested that we would have to get the pixels much smaller in the images and that we would have to make the diodes 1/4000 as small as they are right now to achieve that goal. He also explained that he may not completely comprehend what we are doing and that this might work. He would like to be updated on our progress in the future.

5.1.14 Stan Gordon, President of Cortex Medical Management Systems Inc., Aug. 29, 1995

The name of Cortex Medical Mgt.'s pathology module is *The Gold Standard.* It is mainly a stand alone system, but has the ability to interface to HIS systems. The most common HIS systems the company has interfaces to are SMS and HBO&C. There are 67 installations of this pathology module.

5.1.14.1 Images

The Gold Standard does not handle images yet, but the company is definitely planning on adding this feature to their system. The types of images handled will be micrographs, Mr. Gordon said. In addition, the images will be digital and will be in color. The images will also be used mainly for patient filing and not for reference. The software the company uses right now is canned, not in-house.

Mr. Gordon explained that their product is primarily in DOS right now, but will be switched to Windows by the fall. (They hope to have a beta site by the fall).

The storing capability of the module is unlimited. Mr. Gordon explained that they will use CD jukeboxes for storage, which usually does not have a limit to it. The method of indexing and file structure will be a sequel server and will be built by them. When asked about the retrieval time of an image, Mr. Gordon estimated that it would take no longer than about five seconds. In addition, the company will store the images permanently.

The images will be able to be shared between pathologists, and probably done so through an image file server. Mr. Gordon said that the networks would probably be EtherNet and Microsoft NT. Both wide area networks and local area networks will be used, but the majority will be local area because that is what most of their clients use.

Mr. Gordon did not know the data rate of the network.

5.1.14.2 Voice

Mr. Gordon did state that their system will have voice by the fall (when they get their beta site). It will be voice recognition.

5.1.14.3 User Interface

The Gold Standard utilizes a mouse and a keyboard.

5.1.14.4 Host Computers

The platforms they support and microprocessors they use are Intel. The 586's will be the standard. As far as the memory on the hard drive and RAM, Mr. Gordon said it was probably about 300MB on the local and 2GB on the file server.

5.1.14.5 Cooperation

In developing their system, Cortex works mainly with pathologists, their primary source of business.

The company has two installations in Arizona. One was just recently installed at University Medical Center (UMC) here in Tucson. The other is down in Sierra Vista. Regarding the installation at UMC, Mr. Gordon provided me with the name of a cytogeneticist who is most familiar with their system - Mark Stevens. He was the head of the installation team at UMC.

Mr. Gordon is very interested in working with us as we develop our workstation. A lady by the name of Judith Krebs will be coming down to Tucson on Sept. 18 to check on the system at UMC. She is the Director of Installations for Cortex. In addition, Mr. Gordon gave me the names of two more companies whom he thinks might be interested in this and whom he is interested in working with. They are **Dianon** in Bridgeport, Conn. and **Neopath** in Seattle, WA. Dianon is very involved in information systems and also very interested in imaging. Mr. Gordon suggested trying to get a hold of their number and giving them a call. Neopath works mainly with Pap Smears. Grace Bartu is the name I was given to contact. She is working on her Ph.D. in Alzeimers and is the principle scientist at Neopath. Her number is (206) 455-5932.

Mr. Gordon requested three more copies of the letter and article initially sent to him. In addition to those, I sent him two copies of Boeckeler's preliminary data sheet and two copies of the information about Kensal which will appear on the Web. Mr. Gordon will be sending literature on their system soon.

5.1.15 Rob Deal, Antrim Corporation, Sept. 11, 1995

Antrim's pathology module is called the *Anatomic Pathology System.* It can be a stand alone system or interface to either HIS systems or LIS systems. When interfacing to LIS systems, it is usually one of Antrim's own systems. The company interfaces to all the major HIS systems, and has a total of 73 installations.

5.1.15.1 Imaging

Antrim does not currently handle images. Mr. Deal stated that their is not a big call for it by their customers (due to expected costs he thought). Though they are interested in possibly handling images down the line, it is not planned for the near future. It is something he is keeping in mind, however. If the company does eventually pick up images, they will be microscopic and gross images. Mr. Deal said that they do not work with radiology or X-rays.

5.1.15.2 Voice

Mr. Deal said that the *Anatomic Pathology System* can have voice. It can have speech recognition or speech response.

5.1.15.3 User Interface

The *Anatomic Pathology System* utilizes primarily a keyboard. They can use a touch screen, but their is not a big demand for it.

5.1.15.4 Host Computers

The platforms supported by Antrim are DEC VAX, Alpha, IBM Risk 6000, and Hewlett-Packard DHP 9000. Their microprocessors are solely Intel.

The hard drive memory is from 1GB and up, Mr. Deal speculated, and the RAM is 32 MB and up.

5.1.15.5 Cooperation

Antrim works primarily with their customer base, mainly laboratories, in the development of their systems. Since they sell their product to customers who are more stand alone, they don't work with a lot of hospitals.

Mr. Deal could not think of any AP system installations in Arizona.

In as far as working with us in the development of our system, Mr. Deal did not want to make any sort of commitment yet because it is not a focus of their immediate future. He would like to be kept updated on what we are doing however, so that if and when they decide to carry images, he can have a reference to call on.

5.1.16 Nancy Oakland, Marketing Manager of Health Sciences Systems, Sept. 25, 1995

The name of Health Sciences Systems (HSS) pathology module is *OPUS*. It can be a stand alone system if sold with the clinical package, or it can be interfaced to HIS systems. Ms. Oakland stated that as far as interfacing goes, they are HIS compliant. HSS currently has 25 installations, all in hospitals of varying size.

5.1.16.1 Imaging

OPUS does not handle images, and there are no plans for adding that feature in the near future. If and when images are implemented, Ms. Oakland did not know what kind of images they would be (i.e. microscopic, gross, or X-ray). She guessed that there was a 50-50 chance the images would being digital or analog, and she was almost certain they would be color.

Ms. Oakland also speculated that the images would be shared between pathologists. She said that if anyone wants to sell their product, they would almost have to go in that direction because that is the direction the technology of hospitals is going. The company would probably use an image file server for this purpose, and a wide area network.

5.1.16.2 Voice

The *OPUS* system will have voice. They are currently working on voice recording, voice recognition, and synthesis.

5.1.16.3 User Interface

Currently the module utilizes a keyboard, but the mouse and voice are right around the corner, Ms. Oakland said.

5.1.16.4 Host Computers

HSS supports mainly the UNIX 9000 platform, as well as their microprocessors. They do occasionally use the IBM AS/400 type.

The storage capability of their system is dependent on the hardware they link to. Ms. Oakland explained that with the UNIX HP900, you can fit 32,000 patients on it before you have to move to optical disk archiving. So their capacity is quite large.

5.1.16.5 Cooperation

HSS has worked with one large group over the past eight years in developing their system. They have developed their system for the needs of this group, and with this co-partnership, Ms. Oakland explained, their company has been able to grow into a large reference lab. Also, she is currently trying to get an agreement with the University of Florida to install their system into the Veterinary Medicine department.

They do not have any installations in Arizona, but are actively looking for one.

Ms. Oakland stated that although they are not in any position to work with us right now or in the near future, she would like to be kept updated on our progress. She will send me some literature on their pathology module for me to review. She also stated that if she could ever be of any additional help to please contact her.

5.1.17 Dr. Selig Leyser, EasyPath Software, Sept. 29, 1995

Easy Path is the name of EasyPath's pathology module. It is a stand alone system with three installations in the U.S. One of the installations is at Baylor University in Houston, Texas, while another is at the Fred Hutchison Center in Seattle, Washington (this is a center for bone marrow studies). The installations are solely databases at the moment. Though not active, the module may also have some Meditech integration.

5.1.17.1 Imaging

Easy Path can support full colored images. The kinds of images carried are microscopic (cytology and surgical), and are typically used for both reference purposes and as images for patient files. The images are digital and either color or black and white. Dr. Leyser did not know the size of the images or the number of bits per pixels. The software used to handle the images is an in-house software.

The storing capability of the module for images is unlimited for the database. Typically, Dr. Leyser said, three images per case is the standard (this can be changed). The method of indexing employed is through 4th dimension from Windows, which allows indexes to be "on" or "off." The file structure is a relational database with client servers. Dr. Leyser stated that the retrieval time of the images depends on the hardware, and he has done no testing to date to see just how long it takes. The images may be perged at any time, so storage time is determined by the user. Dr. Leyser said that the imaging part of his system is not in great use.

The images can be shared between pathologists. They would be shared over a network (EtherNet), with a hard drive storing all the data. Dr. Leyser stated that there is the possibility for CD-ROM's in the future. A local area network would be used, and the data rate is whatever the rate of EtherNet is (he did not know).

5.1.17.2 Voice

Easy Path does not have voice, but Dr. Leyser is interested in it. He is looking into a system called "Power Secretary," which would conduct voice recognition.

5.1.17.3 User Interface

Currently the pathology module utilizes a mouse and a keyboard.

5.1.17.4 Host Computers

At this time, the platforms supported by Dr. Leyser's system are solely Macintosh. He hopes to be switching to Windows 4-D in the future (a year or so). There is also the possibility of using UNIX. Microprocessors are from Motorola (they are a PC chip for the Powermac, which has a must faster speed than any others).

Dr. Leyser guestimated the memory on the Client Server to be between 3.5 and 5 megabytes, while the server could handle anywhere from 12 to 20 megabytes or more. He also stated the size of our images (25 megabytes uncompressed) would be very impractical for archiving (they would be too big for storing on anything but a CD-ROM, for example).

5.1.17.5 Cooperation

Dr. Leyser does not work with any professional societies or government agencies in the development of his system. He is a practicing physician and works off of what he knows and gathers. He does not have any installations down in Arizona.

Dr. Leyser is interested in working with us as we develop our workstation. He reiterated the fact that his system has compression, has images, and is very economical (approximately $5000.00). He also emphasized that he would be very willing to come down and put on a demonstration of his system for us.

He is sending literature for us to view.

5.2 Pathology Images From The Internet

For future comparison to images produced by Kensal Corporation, several microscopic images were viewed from various institutions with Pathology Image Databases on the Internet. Almost all of the institutions were universities who use the images for education purposes at their medical schools. Although attempts were made to contact those in charge of producing the images, no one responded. Therefore, it is not clear how these images were produced, or at what magnification they were taken at.

Images from eighteen institutions nationally and internationally were viewed. Quality, size, color, contrast and format of each were compared. Out of these 18, only two to three institutions had images whose quality was above average. Two of these institutions were Cornell University and the University of Utah. A brief description of their images follows. The other institutions had images whose quality varied. Blurriness of details was the most common problem, making the tiny structures of microscopic images hard to see. Color was a second problem, as some

institution's images had extreme colors of bright pink or red versus the regular magenta color. In other cases, the color appeared too light, washing out details. With occurrences such as these, images lack a sharp, crisp look to them that make them easy to view. Once again, knowing how these images were produced may help to gain some insight as to why these problems have occurred and how to avoid them in the future.

5.2.1 Cornell University

Cornell University has approximately 126 pathology images, both gross and microscopic, which are stored using the CompuServe GIF Format. Approximately 50 images were downloaded, focusing solely on the microscopic images. The following information was obtained from these images. The images are 8 bits per pixel (indexed color) and range in size from 122K to 179K. The average width and height in pixels is 500 x 300 respectively, and the resolution is 72 pixels/inch. Each image is titled with what the tissue is and where it is from.

Overall, the quality of these images is very good. The color and contrast are sharp, and the details are easily visible on most of the images. Of course, some images are better than others, but a majority of the ones viewed were some of the best images found. Cornell has been e-mailed to find out more about their images and to offer some insight to our future plans with imaging. No name could be found to send the message directly to, and no response has been made as of yet.

5.2.2 University of Utah

The images found on this file are probably the best I have seen so far. Utah uses these images for medical education. Their "electronic laboratory" includes more than 1500 archived images demonstrating gross and microscopic pathologic findings. The images on the file are scanned from kodachromes to make a Photo-CD, and are saved in the JPEG compressed format. They are 24 bits/pixel (RGB color), and range in size from 56K to 489K, with the latter being more common. Resolution is 72 pixels/inch, and the average width and height is 504 x 330 pixels respectively.

The images are filed under *subject areas* (11 areas total including cardiovascular, pulmonary, GI, renal, dermatopathology, hematopathology, neuropathology, forensic pediatric-perinatal, reproductive organ, and clinical pathology), *mini-tutorials* (7 total including such things as AIDS Pathology and Pathology of Drug Abuse), and *organ systems* (9 systems such as bone and joint, breast, and endocrine). The number of images under each category ranges anywhere from 5 (smallest) to 53 (largest) images.

Twelve images have been viewed mainly from the *subject areas* (two from the *mini-tutorials*). As mentioned above, they appear to be of great quality. The color and contrast are sharp, and details are very clear. All of the images seen have been obtained from autopsies.

6. NEO LENSMAN

This section briefly describes the functions and operation of the application Neo Lensman (hereafter Lensman) and a method for calibrating light. Most functions are executed from floating windows called windoids (from the 'Inside Macintosh' series describing floating tool palettes). All windoids from Lensman are illustrated.

6.1 Function Description

6.1.1 Main Floating Window Description

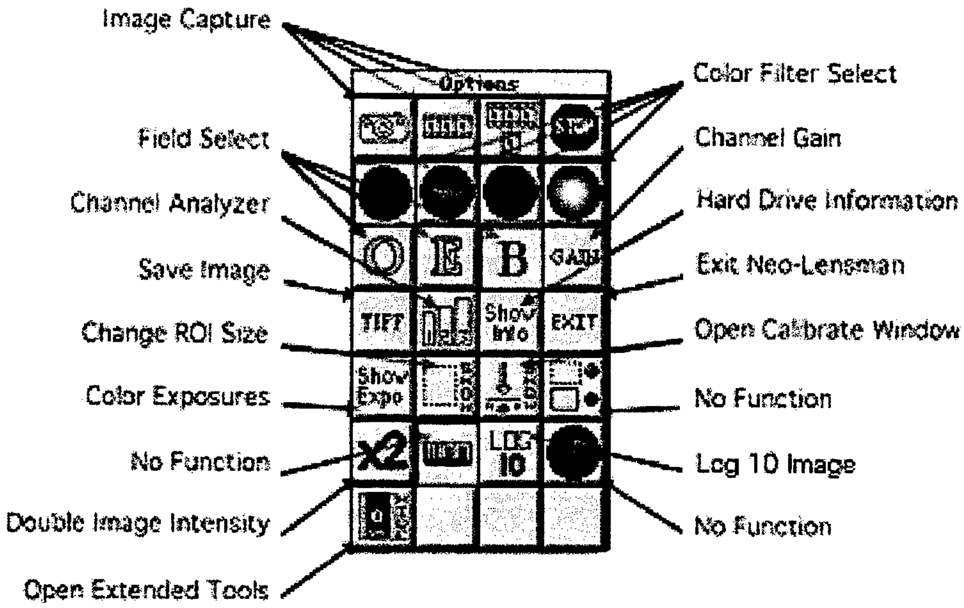


Fig. 1 - Main Floating Windoid.

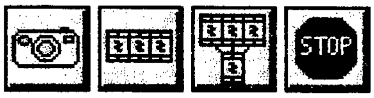
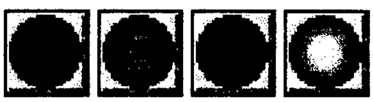
6.1.1.1 

Image Capture

The Image Capture commands begin and end polling the camera for image data and display the results on screen. The first command takes only one picture and displays it on screen*. The second command continuously takes pictures and displays them until the "Stop" button is pressed. The third command continuously polls the camera for pictures and integrates images for improved quality. The last command, "Stop", stops polling the camera for images.

* In the latest version of Neo Lensman, the single image capture (the button with the camera icon) is nc operating properly. Pressing the button seems to acquire something from the camera, but does not display an image.

6.1.1.2  *Color Filter Select*

Pressing a color button sets the camera's filter wheel to either red, green, blue, or clear. Listen for the changing filter wheel in the camera to be sure it actually has changed. A color button is always depressed (selected).

6.1.1.3 

Field Selection

The "Odd", "Even", and "Both" commands set the software to only capture either the odd, even, or both fields of the image. To speed picture taking you can set the camera to capture either the even or odd fields only. Capturing just an odd or even field effectively halves the image vertically. To capture both fields, i.e., a full frame, the "Both" command must be set.

6.1.1.4 

Channel Gain

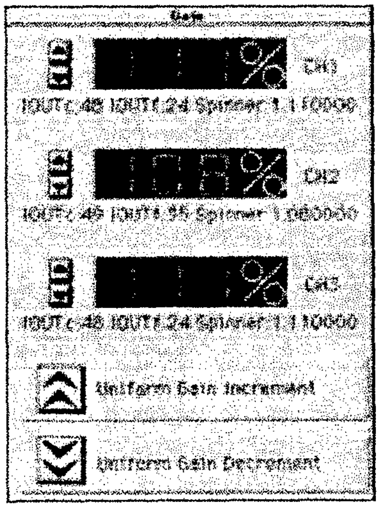


Fig. 2 - Gain Windoid.

The "Gain" function sets the gain, individually or as a whole, of the three channels. Figure 2.104 shows the "Gain" windoid with three displays with corresponding spin controls, and uniform gained incremen or decrement. Clicking this button again will close this windoid.

6.1.1.5 

Save Image (TIFF)

Opens the Apple standard dialog for saving the currently displayed image in a TIFF format. After the image is saved it can opened by any application that can read 8-bit TIFF images.

6.1.1.6 
Channel Analyzer

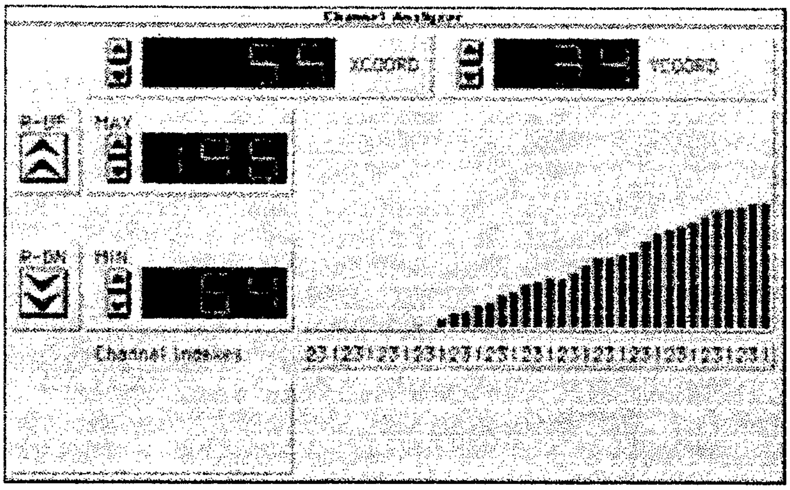


Fig. 3 - Channel Analyzer.

Within the image window is a small flashing blue line whose coordinate location is given by the displays in the upper portion of the "Channel Analyzer" windoid (here 55,34). The intensity values along this line are displayed in the "Channel Analyzer" windoid by red bars. Selecting a set of channels (red bars) will highlight them, visually emphasizing a group of channels. The upper and lower limits of the red bars are shown and altered with the left "Min" and "Max" displays. Using this windoid in conjunction with the "Gain" windoid allows the use to equalize the channels. Clicking the "Channel Analyzer" button again will close this windoid.

6.1.1.7 

Hard Drive Information

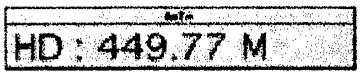


Fig. 4 - Information.

All available space (in Megabytes) on the current hard drive is displayed in this windoid. Clicking the "Show Info" button again will close this windoid.

6.1.1.8 

Exit Lensman

Terminate the current session of Lensman.

6.1.1.9 

Display Exposure Windoid

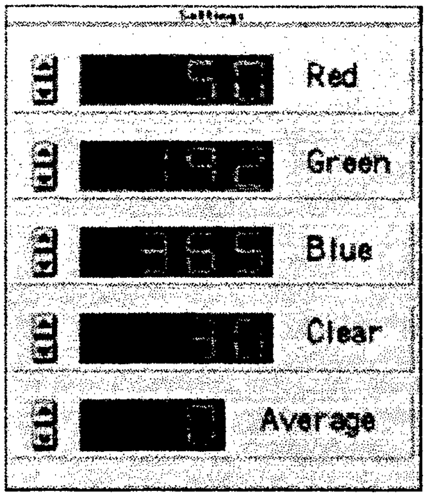


Fig. 5 - Settings Windoid.

From within the "Settings" windoid (exposed with the "Show Expo" command) the individual exposure settings for each color filter are set. Clicking the "Show Expo" button again will close this windoid.

6.1.1.10  *Alter ROI Size*

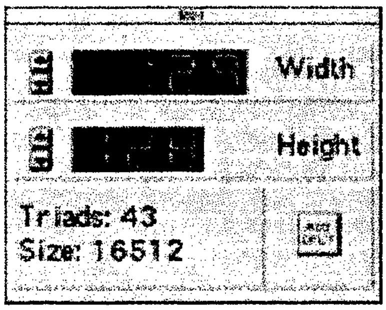
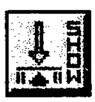
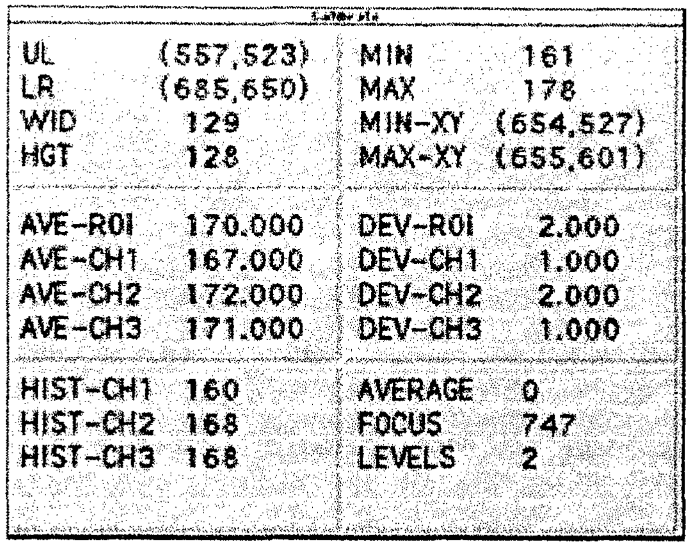


Fig. 6 - ROI Size.

Within the image window is a region outlined by a moving marquee . The region is located at the cente of the image. All measurements are taken from within this region. The "ROI" windoid sets the width and height of the region. The default settings are returned with the "ROI DFLT" button in the lower right of the windoid. Clicking the "Show [Region]" button again will close this windoid.

6.1.1.11  *Display Calibration Window*



UL	(557,523)	MIN	161
LR	(685,650)	MAX	178
WID	129	MIN-XY	(654,527)
HGT	128	MAX-XY	(655,601)
AVE-ROI	170.000	DEV-ROI	2.000
AVE-CH1	167.000	DEV-CH1	1.000
AVE-CH2	172.000	DEV-CH2	2.000
AVE-CH3	171.000	DEV-CH3	1.000
HIST-CH1	160	AVERAGE	0
HIST-CH2	168	FOCUS	747
HIST-CH3	168	LEVELS	2

Fig. 7 - Calibration.

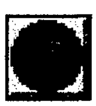
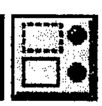
The "Calibrate" windoid lists a series of measurements. All measurements are listed in Figure 7. Clicking the "Calibrate Icon" button again will close this windoid.

6.1.1.12  *Double Intensity Values*

The "2x" command allows the user to select an already saved TIFF file (with the Apple's standard "Open" windoid") and double all pixel values. Lensman saves the changes in it's own folder with a "2_" in front of the filename.

6.1.1.13  *Log 10 Function*

The "Log 10" command allows the user to select an already saved TIFF file and perform a log operatio on all pixel values. The changes are saved in Lensman's folder with a "L_" in front of the filename.

6.1.1.14    *Non-Functioning Operations*

The commands marked "No Function" in Figure 1 are reserved for future use or are not operating properly. The three open buttons at the button of the Main Floating Windoid are also reserved for future use.

6.1.1.15  *Open Extended Tools*

This command opens the extended tools under the "Test" windoid. The extended tools are listed and described in Section 6.1.2. Selecting this button again will close the extended tools.

6.1.2 Extended Tools Description

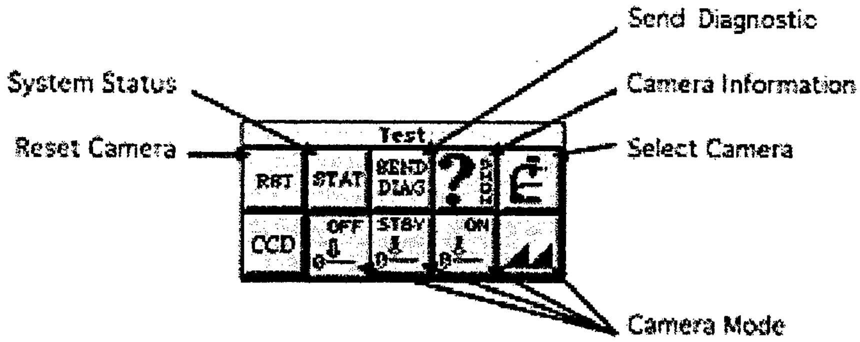


Fig. 8 - Extended Tools Windoid.

6.1.2.1  *Rest*

Resets the camera and provides the option to return all values (gain, exposure, etc.) to default values.

6.1.2.2  *System Status*

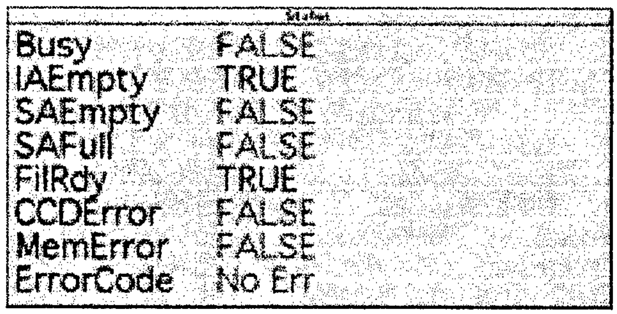


Fig. 9 - Camera Status

The "Stat" commands returns a list of Boolean values in the "Status" windoid. The list of operands are shown in Figure 9. Clicking the "Stat" button will close this windoid.

6.1.2.3  *Send Diagnostic*

The "Send Diag" command sends a diagnostic operation to the camera to check for errors. The operation usually takes about two minutes to complete.

6.1.2.4  *Camera Information*

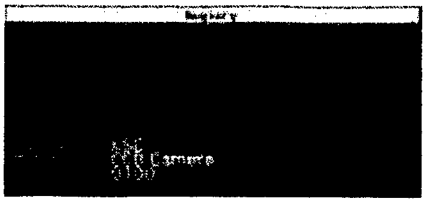


Fig. 10 - Camera Information

Figure 10 displays the "Inquiry" windoid with listed information. Clicking this button again will close the "Inquiry" windoid.

6.1.2.5  *SCSI Camera Select*

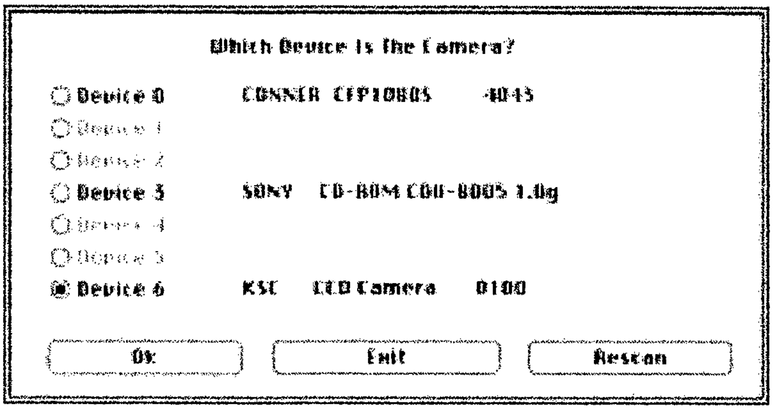
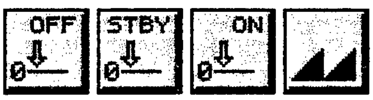


Fig. 11 - Camera Select

This command displays a dialog asking the user to select the camera. All seven [available] SCSI devices are shown. Clicking "OK" sets the current selection. "Exit" will exit Lensman. "Rescan" will scan the SCSI bus again.

6.1.2.6 

Unknown

6.1.2.7 

Camera mode

6.2 Menu Commands

6.2.1 About Neo Lensman (under "Apple" menu)

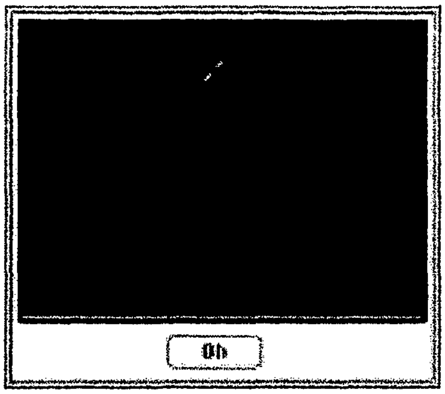


Fig. 12 - About Lensman

The "About Neo Lensman" command opens a dialog box describing Kensal. Clicking the "Oh" button closes the dialog.

6.2.2 Quit (under "File")

Exit Neo Lensman

APPENDIX A
RECOMMENDED LIGHT CALIBRATION

What follows is the recommended method for calibrating light on the microscope and within Lensman:

1. Insert slide and move to clear glass.
2. Set the desired Objective (this process must done for each objective) and focus visually.
3. Set the microscope slider to 7.
4. In Lensman, set the filter to RED.
5. Open the "Calibration" and "Exposure" windoids.
6. Begin polling the camera continuously.
7. Adjust the microscope field stop until the "circle" is just outside of the captured image.
8. Adjust the RED exposure in the "Exposure windoid until all "Hist Chn"'s in the "Calibration" windoid reach the value of 120. DO NOT adjust the microscope's light slider.
9. Switch to the GREEN filter.
10. Adjust the GREEN exposure in the "Exposure" windoid until all "Hist Chn"'s in the "Calibration" windoid reach the value of 120. DO NOT adjust the microscope's light slider.
11. Switch to the BLUE filter.
12. Adjust the BLUE exposure in the "Exposure" windoid until all "Hist Chn"'s in the "Calibration" windoid reach the value of 120. DO NOT adjust the microscope's light slider.
13. Close the "Exposure" window.
14. Make sure the "Calibration" window is still open.
15. With light calibrated, move to a desired location.
16. Set the filter to GREEN
17. Focus visually through the oculars.
18. With the camera continuously acquiring images, focus the image on screen.
19. Now focus the microscope stage until the "Focus" number, in the "Calibration" windoid, peaks.
20. The camera in now focused for this location. Each time stage location changes (especially at higher magnifications), refocus the camera.
21. To end polling the camera continuously select the "Stop" button.

With the camera calibrated for the current objective, and focused for the current location, three images can be acquired to create a color image:

1\. Begin polling the camera continuously.
1\. Set the filter to RED while continuously acquiring images.
2\. After a RED image is display, select the "TIFF" button to save the image to disk.
3\. Set the camera to the GREEN filter while still capturing image continuously.
4\. After a GREEN image is displayed, save to disk with the "TIFF" button
5\. Set the camera to the BLUE filter, and after an image is acquired, save to disk.
6\. In another application, such as Adobe Photocopy, these three images can be combined to create a 24-bit color image.

7. SOFTWARE FOR POWERMAC (by Gregory Guerin)

Kensal Corporation's Triakis program contains a software morphology engine written in 68000 assembly language. It was my goal to convert this to run on the PowerPC-based Macintosh, and benchmark its speed. The strategy I chose was to convert the 68000 assembly language into a high-level C form, compile that C code for the PowerPC, then evaluate the resulting code, both for actual speed and overall quality of compiler translation. Other Triakis functions written in 68000 assembly language and converted to run on the PowerPC-based Macintosh and benchmarked for speed are the workspace booleans operations, boundary-setting and volume operations, and the encoding of a gray-scale image into a workspace solid.

To make the conversion process more tractable, Kensal representatives agreed that only the "mode 0" portion of the assembly-language engine would be converted for the initial benchmarks. Since this was believed to make up over 99% of all current uses of Triakis, it was deemed to cover most existing customer uses.

After analysis and conversion of the 68000 assembly language into a C version, the new C code would be validated first on a 68000-based Macintosh, before recompiling for the PowerPC. The C engine's output would be compared byte-for-byte with the output of the known-working 68000 assembly-language engine, after feeding identical inputs to both routines. If both the C-code and the assembly-code routines produced identical output for a variety of identical inputs, then we could conclude with a fair degree of certainty that the new C code was correct. It could then be recompiled for PowerPC and benchmarked for speed and C-compiler code quality.

7.1 Validation Results

These results are in the file "=PowerTest (68K).out.68K". The program (included in package) that produced this file contains working versions of both C and 68000 assembly-langauge versions of the morphology engine. The assembly-language was taking verbatim from source code supplied by Kensal Corp, with minor editing done to make a C-function prolog and epilog so that a proper C stack-frame was correctly set up. As a further check, the actual executable code was disassembled from the new program, and compared with a disassembly of known-working code from a Triakis test-facility program. Since the code matched, it was deemed correct in its new assembly-language form.

The C code was written so as to improve PowerPC execution time, not necessarily optimal 68000 execution. In particular, some idiomatic expressions C tend to produce sub-optimal PowerPC code, so these were avoided in favor of generally simpler expressions, or expressions which did not depend on just-calculated results, thereby exploiting the PowerPC's super-scalar pipelined architecture. By far the best use of PowerPC resources is to have the C code arranged so that everything can be placed into PowerPC registers. This technique quickly exhausts the available 68000 registers, but we were not trying to create the best C-code for the 68000, merely a correct and verifiable representation that would produce good PowerPC code.

The test-bed program (known as "PowerTest") created for this exercise performs some trivial transformations on mechanically generated data. The data for the required look-up table (LUT), shift-table, and mask-table were determined by substituting a "dump-to-file" replacement routine into Kensal Corp's existing Triakis test-program. This data made sense, once it was seen, so it was used to create "test-tables" for the PowerTest program.

The PowerTest program runs the two morphology engines (C and 68000 assembler) over several input data sets designed to exercise the main data-dependent pathways of the engine routines. It then performs a byte-for-byte comparison of the output of these two routines, and

emits a unique message if it detects a difference. It performs these tests over various sizes of data sets, representing typical ranges of user data.

The morphology engine itself is written in such a way that certain commonly occuring data patterns are recognized quickly, then handled trivially. All other data patterns are handled by the slower but fully generalized part of the engine. The two "trivial cases" are patterns of all 0-bits and all 1-bits. All other bit-patterns are passed to the general-purpose portion of the engine. These three pathways are the primary data paths through the engine. After this initial 3-way decision, a single intermediate value is produced, which is then handled in a uniform manner to generate an output value. Hence, if we generate input data that matches each of the "trivial" cases, as well as the "everything else" case, we can expect to exercise all relevant pathways in the engine. Furthermore, if we observe differences in output between the known-good 68000 assembly-language engine and the C engine, we can narrow down the scope of where the error might be by observing those patterns that fail.

The C engine validated perfectly when subjected to input patterns of all 0-bits, all 1-bits, and alternating bits (hex 55). It also validated other test patterns fed to it, including count-down patterns, and count-down with mask patterns. These patterns were validated for data sets consisting of cubes having edge lengths of 64, 96, 128, 192, and 256 voxels. All data matched perfectly between the C and assembly-language engines.

A side effect of this validation process is that we can see the relative speeds of the C and assembly-language engines. Running on a 68040 (40 MHz, Quadra 840AV), the assembly-language engine is consistently faster than the C engine, by about 30%. That is, the assembly-language engine takes only about 70% of the elapsed time of the C engine. This varies slightly depending on the data, but not by much.

7.2 Interpreting the Results Files

Each result file is the output of a complete run of one of the test programs: PowerTest for 68K or PowerTest for PPC. Each test run feeds various test data through different sizes of cubic data sets, measures and displays the elapsed time for ONLY the engine-processing phase, and then moves on to the next data set or size. That is, the time to allocate or fill in the test data, or any other overhead beyond actually running the morphology engines is NOT included in any displays of elapsed times.

The C code determines (through compile-time conditionals) whether it is 68000 code or PowerPC code, and whether it was the CodeWarrior or MPW C compiler. (During development, the MPW C compiler was used as a validity check against the CodeWarrior compiler. The MPW results are not included here, since they are not relevant to the PowerPC benchmarks.)

At run-time, the code also determines the machine ID, and both the virtual and native CPU identifiers. This information appears at the top of each cube-size series of tests. Machine ID 78 is a Quadra 840AV, run by a 40 MHz 68040 CPU, which has 4K of on-chip code cache and 4K of separate on-chip data cache. Machine ID 65 is a Power Mac 8100/80, run by an 80 MHz PowerPC 601 CPU, which has 32K of unified (code+data) on-chip cache, and 256K of off-chip level-2 cache. The Power Mac has a "virtual CPU" of a 68020 without an FPU, which is the CPU model for "emulated" software (hence Apple's designation of a "68LC040 emulation" isn't exactly true, but is not relevant here, so will not be discussed further, except to note that the emulated 68000 code ran about 3 times slower than the Quadra 840AV).

After the information about the machine is displayed, a summary of the cubic data sets is displayed. This is primarily debugger-level information, but it is interesting to see the "perCube" value increase to a healthy 2M for the 256-edge size.

The first calls to the engine routines are to make sure that the Mac's on-demand code-loading process has loaded the routines into memory, so we are not benchmarking disk-read times.

The first three tests are run on a single plane of the cube, as a quick validity check. The term "solid-fill" means that the input data-set is filled completely with the given 32-bit pattern. Later, a "masked-fill" test is done, in which a constantly decrementing counter is masked with the given pattern, and that changing value is used to fill the data-sets. "Masked-fill" tends to produce data-sets with fewer set bits, thus more closely reflecting the nature of actual data.

Since only one plane of data is transformed for the first tests, the elapsed times are not especially representative. The one thing to note is the absence of any messages expressing a difference between the output data-sets.

The next three tests are solid-fill tests over the entire cube of data. These are the speed benchmarks of primary interest. The "all 0's" pattern is trivial case #1. The "all 1's" pattern is trivial case #2. The "$55" pattern is alternating 1's and 0's, and hence always takes the general-purpose pathway through the engine. These three tests demonstrate the expected best-case (all 0's) and worst-case (all $55's) performance. Typical performance will lie somewhere in between.

The last four tests for a given cube-size are intended to show "typical case" performance. They are all "masked-fill" patterns, which means that the given mask is ANDed with a 32-bit down-counter value, and the resulting 32-bit pattern is written to the data-set. The entire data-set is filled in this way. The down-counter starts at a count representing the number of 32-bit values in the overall data-set. So, a 32K data-set would start its counter at 32K/4 or 8192. By varying the mask-value, we can produce data-sets with fewer or greater numbers of varying bits in the pattern.

The mask-value of 0 effectively clears all the down-counter bits, so the pattern is identical to a solid-fill pattern of 0, and we observe that the execution times are the same as for the earlier test of just such a pattern. The mask-value of all 1's does nothing to the down-counter, so it produces the most varying pattern of data. The mask-value of all $55's effectively clears alternating counter bits, so the result varies at about half the rate of the all 1's pattern. The mask-value of $4F produces data with a varying 4-bit pattern in the LS bits, occasionally toggling the bit in the $40 position of the mask. This produces a pattern with 3 bytes of 0's for every byte containing any 1 bits, and probably represents a typical sparse data set. You can observe that the times for each of these mask-filled patterns lies between the above-noted best-case and worst-case performance. You can further observe that the C engine is consistently slower than the 68000 assembly-language engine.

7.3 PowerPC Benchmark Results

These results are in the file "=PowerTest (PPC).out.PPC". They were run on a Power Mac 8100/80, and executed as native PowerPC code. The story is most clearly seen in the run-times of the 256-per-edge cube, compared to the 68000 run for the same cube size. The PowerPC C engine is consistently 3 times faster than the 68000 assembly-language engine running on the Quadra 840AV (which is the fastest 68K-based Mac made). In the case of "all 0's", the performance gap is wider still, about 3.33 times faster than the 840AV. The gap between the C engine on the PowerPC and the C engine on the 840AV the is always well in excess of 4 times faster in favor of the PowerPC.

A key thing to note in these results is the appearance of the "unmatched" messages after each test run other than all 0's. These arise because there is no simple way to call emulated 68000 code from a PowerPC native program (or vice versa). The simplest method for writing code is either all PowerPC or all 68K. Since we had already validated the C engine using the 68K-based

Macintosh, it was redundant to validate it on the PowerPC. So, the simplest way to compile for PowerPC was to define an empty function named "PlaneProcess_68K", which appeared instead of the actual 68000 assembly-language function when compiled for PowerPC. This was all controlled by conditional compilation that detected predefined compiler symbols, and automatically selected the appropriate code for compilation. The reason why we don't see the message for the "all 0's" case is that the output data-area is cleared to 0's before the transform test is run, so if nothing runs, it will compare 0's with 0's and always match.

7.4 PowerPC Code Quality

The CodeWarrior C compiler for PowerPC recognizes the large register-set of the PowerPC CPU, and exploits it with reasonably efficient code generation. The assembly language produced by the compiler is shown in "PPC asm". While it might be possible to somewhat improve this code by manually recoding in PowerPC assembly language, it is very unlikely that any improvement beyond 5% or so would be possible, and that 5% would be hard-won indeed. Conveniently, the high-level C code can be written so that it maps fairly well one-to-one to PowerPC instructions. Although this eschews certain idiomatic C expressions such as "*pp++", it can have a substantial impact on the nature of the generated code. Somewhat unexpectedly, what might appear to be "inefficient C" actually generates very good PowerPC code, because it exploits both the nature of the PowerPC instructions (3-operands), and the parallelism of the CPU. So "good code" is actually a series of expressions that has LITTLE dependence on immediately preceding calculations, and "bad code" is a series of expressions in which the just-calculated values are immediately used in following expressions.

Instruction ordering to reduce scheduling delays in the generated code is about as good as it can get. In the end, the primary difficulty in exploiting even more parallelism of the PowerPC is the inherently sequential nature of the algorithm, in which calculated values are fed through a steadily narrowing funnel of operations, eventually terminating in a single result.

7.5 Summary

The C version of the software morphology engine runs about 3 times faster on an 80 MHz PowerPC than the equivalent hand-optimized 68000 assembly-language runs on a 40 MHz 68040. Although once considered "top-of-the-line", an 80 MHz PowerPC is now considered "about average" in Apple's range of PowerPC Mac offerings, and 120-132 MHz is now the high-end. On the other hand the 40 MHz 68040 of the Quadra 840AV has never been matched or exceeded in any other Macintosh. Although there are several 33 MHz 68040 machines, these can be expected to run about 20% slower than the 840AV.

In a sentence, then, an "average" Power Mac will run the software morphology engine about 3 times faster than the fastest 68040-based Mac ever made. Further, writing the engine in C has no significant impact on performance, since the compiler-generated code is fairly well optimized, given the nature of the algorithm.

7.6 Workspace Booleans

These were very straightforward to recode in C, and equally straightforward to test. Trivial C implementations were written, which operated solely at a byte level. Although simple, these routines suffer enormous speed penalties. Hence, the final routines were written to exploit both 32-bit PowerPC data sizes, as well as loop unrolling.

In my experience, unrolling loops by a factor of 4 usually garners the majority of all unrolled-loop speed benefits. Increasing to 8-level unrolls is rarely worth the effort, or the code

size or complexity. Coupling a 4-level loop unroll with 32-bit data size means that each loop iteration would operate on 16 bytes of data at a time. This seemed to be sufficient.

Since the workspace length is specified as a byte-count, up to 15 extra bytes may also need to be operated upon. These were done using a simple byte-oriented loop, after the main loop had run.

The output of the optimized loops was compared byte-for-byte with the trivial version's output, to verify that no coding errors had been made. The optimized loops easily out-performed the trivial byte-oriented versions, with no differences in output ever observed.

7.7 Boundary-Setting

Triakis often needs to set or clear all the voxels that occupy the faces of a workspace, and it needs to do this quickly.

The basic strategy was to fill one plane completely, then to work serially through the other planes, filling the "side" lines rapidly, then moving through each "line end" pair of bytes. The final plane would then be filled completely. This strategy was deemed to make the best use of the PPC's data cache, since this method would always work through in ascending order of address, never revisiting any areas of memory.

The resulting C code was verified by breaking into the debugger and inspecting the data. All data sizes tested worked perfectly. The performance was very good, even with large workspaces.

The boundary-setting routines were further tested as part of the volume-calculation tests, the next modules converted.

7.8 Volume Calculations

Calculating the volume of a workspace (the number of ON voxels) requires examining every byte in the workspace. This was done by reading a 32-bit value from the workspace, translating it through a "count table" in 8-bit chunks, and summing the resulting values. This was the technique already used by the 680x0 assembly language, but extended to the 32-bit values most efficiently handled by the PPC.

This routine was tested by filling workspaces with all 0's, all 1's, and alternating bits, each time calling the volume calculator. Since the dimensions of the workspace were known, the correct results were trivial to verify.

Additional tests were conducted by calling the boundary-setting routines on the above-listed fill-patterns, and verifying that all the results were as expected. They were.

7.9 Gray-Scale Image Encoding

A gray-scale image can be converted into a solid in a workspace by interpreting gray-levels as "column heights", and filling the space accordingly.

The C routine used the same basic method as the 680x0 routine, with one minor change: removing all multiplies and divides outside the main encoding loop. This was accomplished by recasting the necessary calculations in a fixed-point format that maintained full precision, yet could easily be converted to pure integer form with nothing but shifting. The resulting threshold levels

were then validated by actual comparison with the original calculation, and no discrepancies were found. Thus, the validity of the new calculations was determined.

The basic method of encoding was identical to the 680x0 method, where gray-scale bytes are compared in groups of 8 with the threshold, to construct a single-byte value to be placed into the workspace. The speed of this method was more than enough to encode a 128 x 128 gray-scale gradient into a 128 x 128 x 128 workspace in 183 mS on an 80 MHz 601. From this result, encoding into a 128 x 128 x 64 workspace should take half as long, or about 92 mS, which is well under the constraint of 250 mS initially laid down. The performance should be correspondingly higher on faster 601's, or on 604's.

7.10 Comments

The byte-oriented booleans are astoundingly slow on the PowerPC. Although I have no explanation for this, it may be a combination of poor pipelining, poor cache performance, and other factors. However, since these routines were used only to verify the others, they have no impact on the final product.

All C routines have been tested and validated on a 68040 machine. In some cases, their speed is very close to the assembly-language originals, but in other cases the difference is 40% or more. This is not surprising. However, the performance and portability should make it simpler to integrate these new functions during the porting to PowerPC.

The CodeWarrior C compiler was used for development, then the Symantec C (v 8.0) compiler was used for final testing and tuning. No portability problems were encountered at all, but the speed of Symantec C's code was invariably below that of CodeWarrior. Even Symantec's "most optimized" code never exceeded even barely optimized code from CodeWarrior. With instruction scheduling enabled in CodeWarrior (not even a Symantec option), the speed difference increased even more.

Sometimes the speed differences were insignificant, although they were always visible. For example, the booleans tested at nearly identical speed: 263 mS for Symantec vs. 256 mS for CodeWarrior, a difference below 3%.

In one or two cases, the differences were more substantial. For example, the Encode test was 183 mS for Symantec and 149 mS for CodeWarrior, or almost 20%. Also, the FindVolume tests for Symantec C averaged to about 23.7 mS for a 128 x 128 x 128 workspace, while CodeWarrior came in at about 17.1 mS for identical data, nearly 30% faster.

It is quite clear that the quality of CodeWarrior's code generation is substantially better than Symantec's, especially for non-trivial code. Also note that CodeWarrior produces slightly smaller code, although these differences were always well under 10%, and would probably change if Symantec simply generated faster code.

7.11 Included Files and Programs

=PowerTest (68K) — 68K-version of test-bed program (run it and see)
=PowerTest (68K).out.68K — results of run on Quadara 840AV
=PowerTest (PPC) — PowerPC-native test-bed program (run this one, too)
=PowerTest (PPC).out.PPC — results of run on Power Mac 8100/80
PPC asm — PowerPC code generated by CodeWarrior C compiler
PPC C — portion of C code that generated "PPC asm"

8. FORTY MHZ TC217 CCD CAMERA (by Greg Kline)

The TC217 CCD manufactured by TI is a 1154 by 488 pixel frame transfer CCD with on board frame memory. There are three video output channels each outputting a pixel 180 deg. apart from the other. By employing a transfer method called centroid shift during altering frame readouts a resolution of 1154 x 972 can be achieved. For the rest of this report assume a resolution of 1154 x 972 is used. The TI data sheet for the TC217 spec.'s a max. pixel rate per channel of 7.2MHz or a over all pixel rate of 21.6MHz at this speed a frame rate of only 15fps max. can be achieved. The purpose of this project is to design, build and test a 40MHz serial driver (individual channel pixel rate of 13.333MHz) and confirm by the use of a scope valid pixel shape for digitizing later and good light sensitivity during exposures.

8.1 Circuit Description

Refer to schematics test circuit for 14MHz srg's (Figure 8-1). U4 the TC217 is being driven during the parallel transfer period by standard TI drivers U3 and U6. The parallel transfer period timing does not change.

The new serial signals (SRGS and TRG) are derived from Ken Crocker's timing of U1 alteras EPM7128-10 (Figure 8-2) all timing is based on a 80MHz master clock, parallel timing periods are normal 1.2MHz and 3.2MHz. The three SRGS are 13.3MHz each for a 40MHz pixel rate, TRG although a serial signal it is only active during the horizontal time period for three pulses at 3.2MHz. The SRGS and TRGsignals drive SILICONIX TP0610l P-CHANEL enhancement mode MOS transistors with rDS on of 10ohm and a Vgs of -2.4V the schematic shows the prototype wiring but during testing I discovered the 15pf gate capacitance of the TP0610L was a little much for a single I/O pin of the EMP7128-10 (capable of sourcing 5ma) in future versions 2 I/O pins in parallel will be used this will insure fast rise times through TP0610L. For this prototype a 74AC541 octal driver was installed (not shown on schematic).

Q1 - Q4 are operating at +3.5V and -9.5V with peak currents of 118ma and a 1 third duty cycle so a SOT23 packages will be no problem for actual PC boards. The 0 to 5V signals from U1 are inverted and translated to +2 to -9.5V to drive

U2 and U5 ELANTEC EL7202 high speed mosfet drivers capable of delivering peak currents of 2A into high capacitive loads. U2 and U5 are operating at +2V and -9.5V. SRG1, SRG2 and SRG3 then drive through 56 ohm resisters into the CCD. the input capacitance of the SRGS is a max. of 180pf on this CCD I estimate 120pf of input capacitance therefore R6,8,15 were selected to reduce overshoot and undershoot and to achieve a good crossover point between SRG1, 2 and 3 at SRG inputs to CCD as shown in Figure 8-4. Both U2 and U5 are dip packages, U5 is fine in terms of power dissipation its driving only SRG1 at 1 third duty cycle TRG is very small only three pulses during the horizontal interval. but U2 is driving both SRG2 and SRG3 at total duty cycle of 2 thirds getting close to 700mw although in spec the next revision will break up U2 into to packages paving the way for using 650mw SO8 packages.

8.2 Conclusion

Driving the TC217 CCD at a serial register rate 13.33MHz per channel and an over all pixel rate of 40MHz will work. It will provide a sampling area of about 9ns to14ns of valid video.

Figures 8-3 through 8-6 show the SRGS and video output signals at normal speeds of 7.2MHz and then at 13.3MHz.

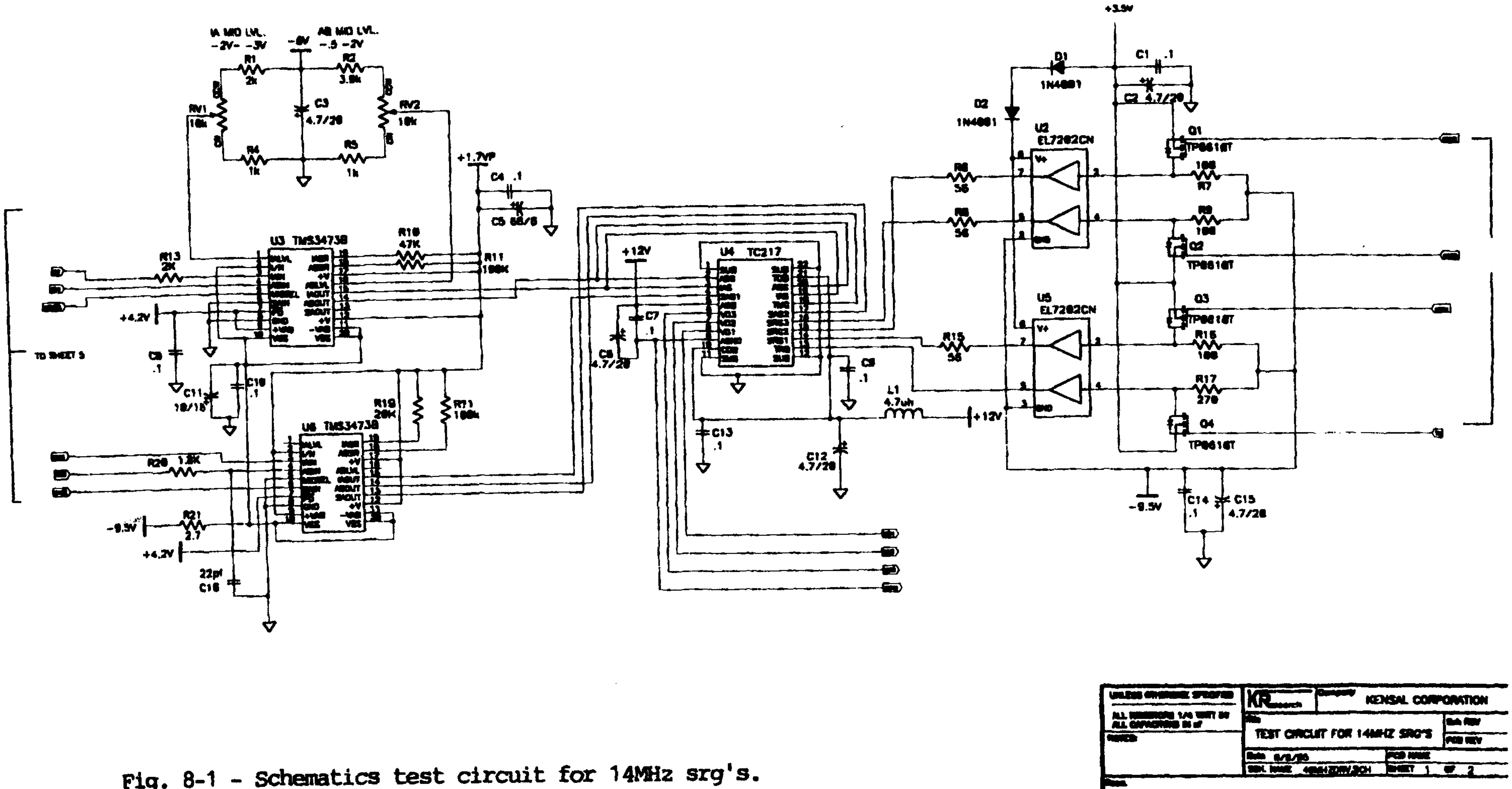


Fig. 8-1 - Schematics test circuit for 14MHz srg's.

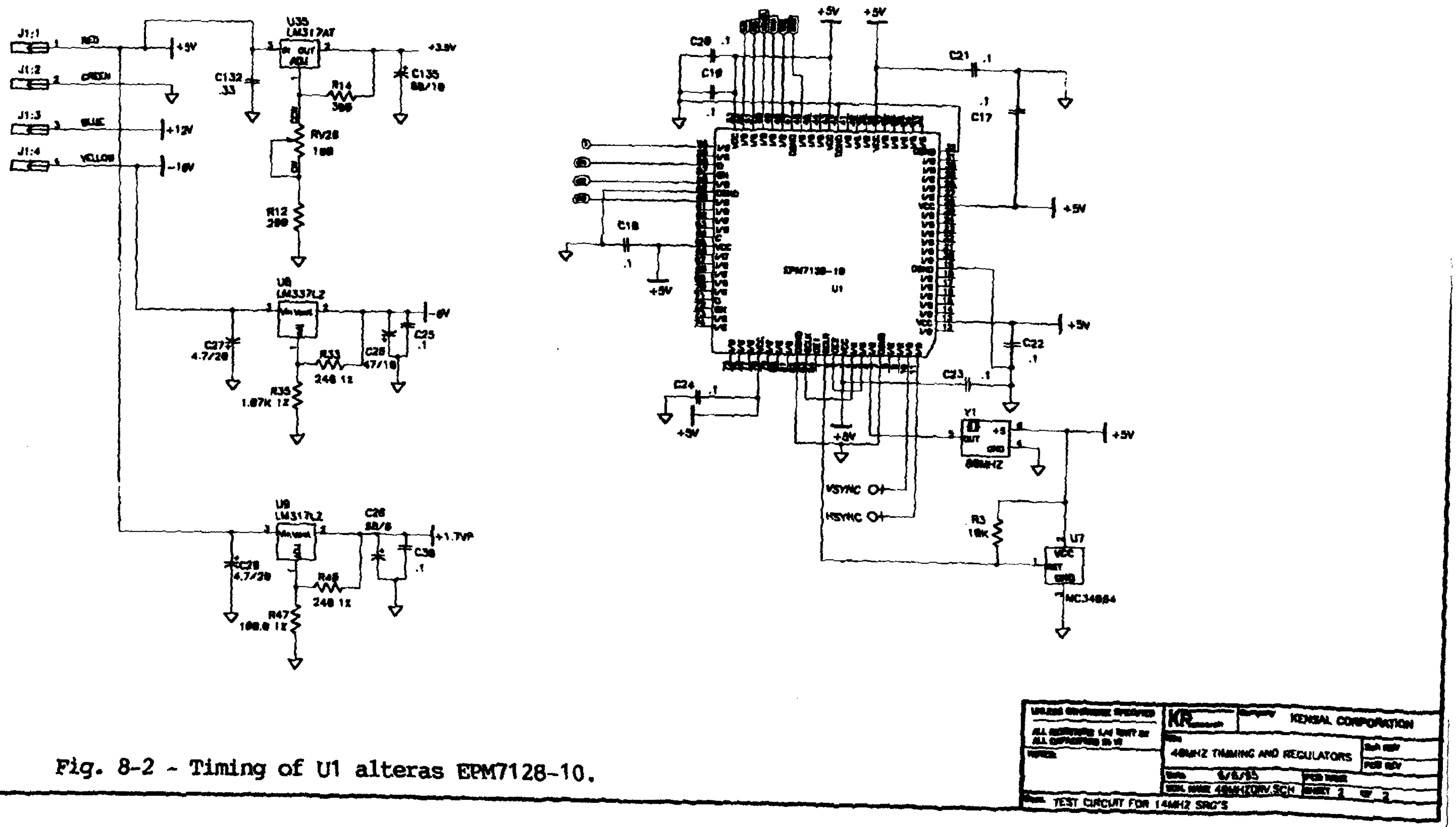


Fig. 8-2 - Timing of U1 alteras EPM7128-10.

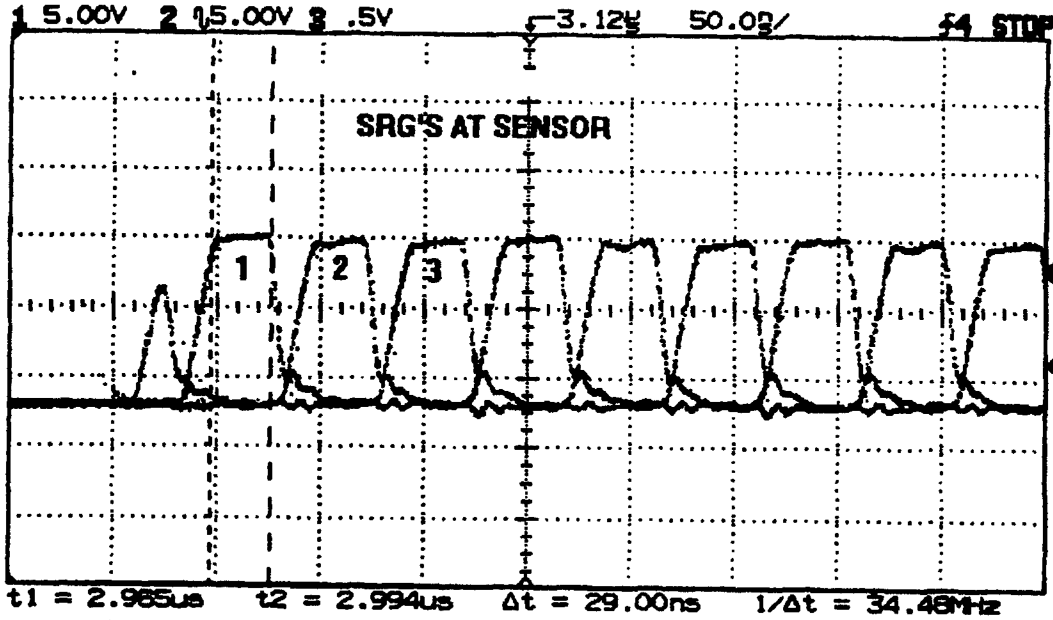


Fig. 8.3 - SRGS at normal pixel rate of 7.2MHz per channel.

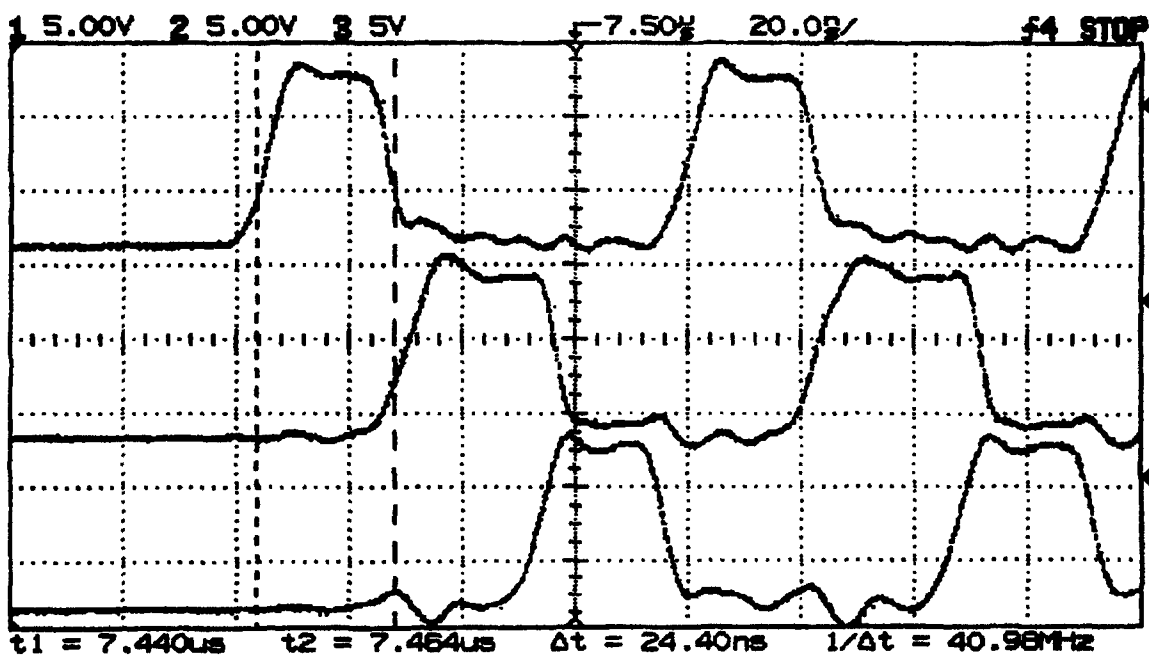


Fig. 8.4 - SRGS at CCD 13.3MHz.

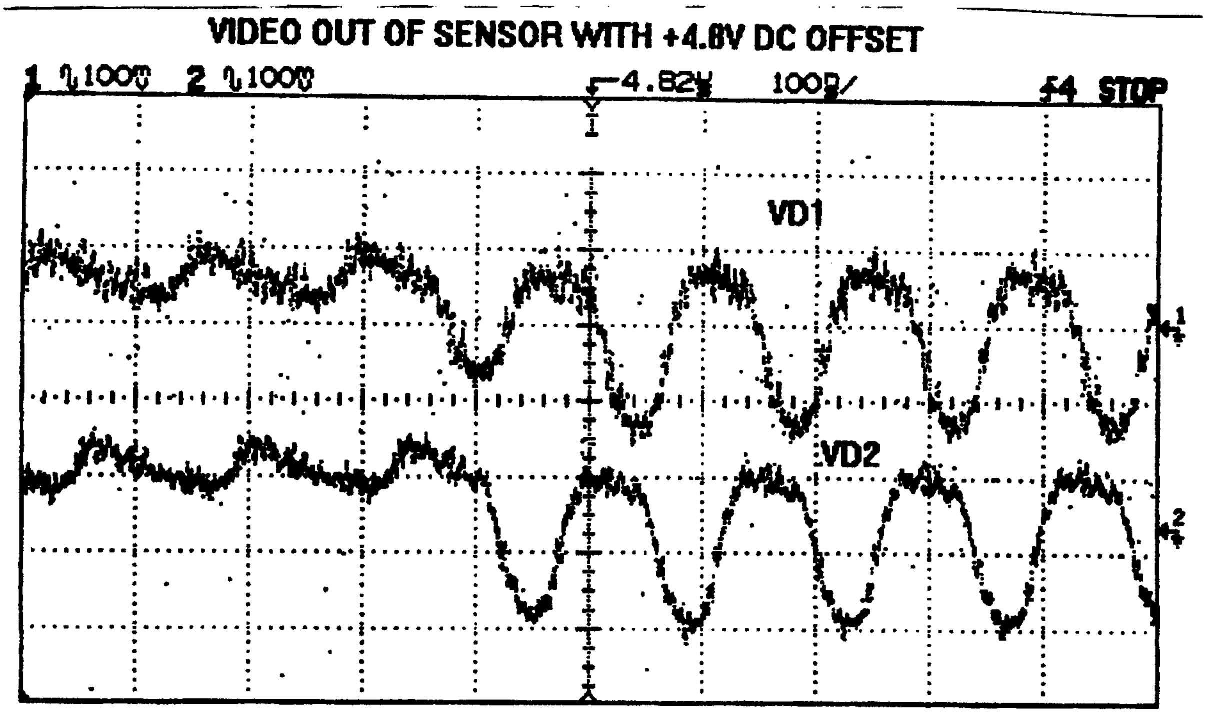


Fig. 8.5 - Normal video out of CCD at 7.2MHz.

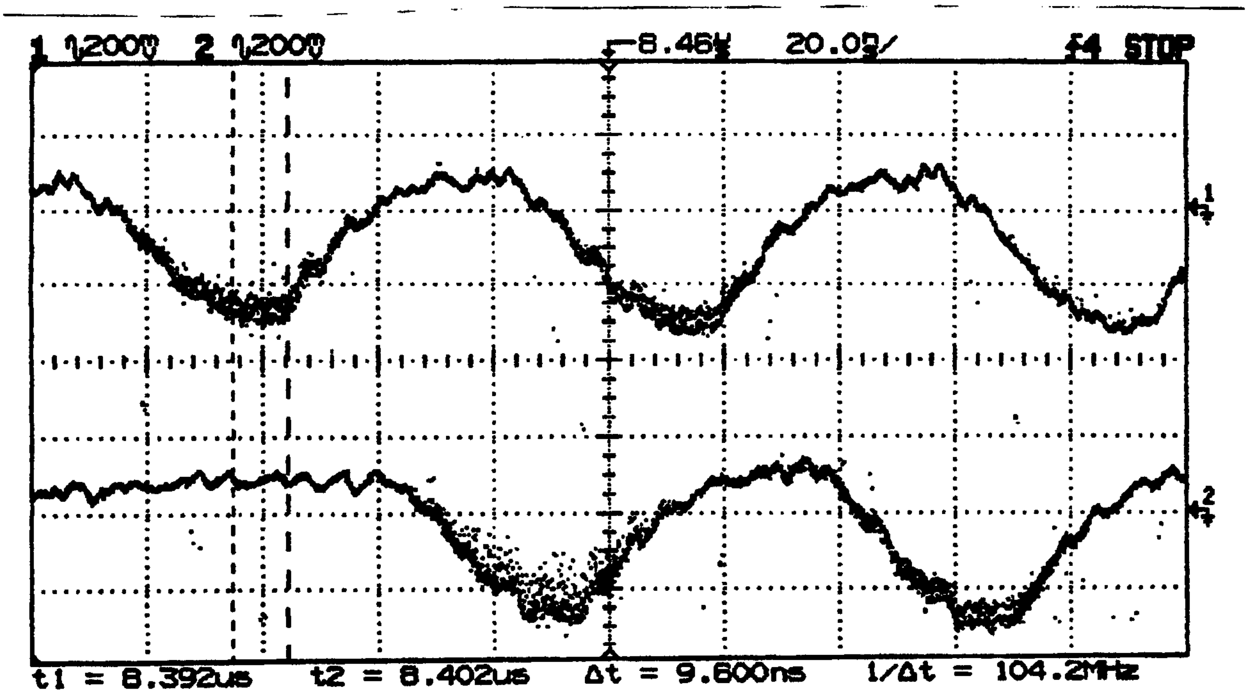


Fig. 8.6 - Video out at 13.3MHz.

9. CONCLUSIONS

This is the final section of our first Annual Report to the U.S. Army Medical Acquisition Activity (Ft. Detrick, MD). It provides four subsections that relate directly to the four areas of activity outlined in Section 1. Recommendations for work to be completed during the second year of our research and development study are given as appropriate.

9.1 Working Environment

The worst mistake that the research and development engineer can make is to be unfamiliar with the environment in which the system he/she is designing must operate. Since PCM (PC Microscope) is an advanced instrument whose use in anatomic pathology is a total departure from all previous microscopes, it was essential that we work with anatomic pathologists (both military and civilian) in order to obtain detailed reactions to our concept. This was done by extensive meetings with the pathology community as reported in Section 2. We found that the majority of pathologists are still apprehensive about computers and their deployment in pathology imaging. (This is totally different from the radiology community where the computer has been an accepted tool since the introduction of computer aided tomography.) Thus conflicting comments were received. The more computer-experienced pathologist accepted our ideas readily, whereas the "old guard" felt that there was no place for computer imaging in the pathology laboratory. Many ideas were advanced concerning the method of viewing human tissue samples at various magnifications using PCM. Our conclusion was that the negative reaction of some pathologists to video display was due to the fact that presently NTSC video is employed yielding a small, low-resolution image. PCM will circumvent this by using HDTV (High Definition Television). This results in an image that encompasses four times the area normally observed under NTSC and where the resolution of the display matches the resolution of the microscope optics.

Furthermore, PCM will use "lensless scanning" to form an image of the full coverslip - a task that cannot be performed by any known microscope system due to the limitations of physical optics. This feat is achieved by the fiber-optic-coubled LSDA (Line Scan Diode Array) furnished by EG&G Reticon (Sunnyvale, CA) with whom we have been working since 1992 when lensless imagery was first demonstrated. The reaction of pathologists to the lensless images that Kensal were able to produce was extremely positive. Some pathologists could even <u>diagnose</u> from the lensless image without using high-resolution except as a confirmation tool.

<u>Future plans</u> will cause the focus of our work on the medical pathology environment to be limited in year 2 to an interaction between the pathology staff at the Luke Air Force Base hospital and Mayo Clinic (both in Arizona) as soon as the Boeckeler prototypes are installed at those two locations and are in operation in a telemedicine network. In this mode either Luke or Mayo will transmit a lensless image from one to the other and the recipient will ask the sender to produce high-resolution pictures of certain areas marked electronically on the lensless image. These high-resolution images will be sent back over the ISDN data link to the requester. Next, the requester will perform his/her diagnosis. This will be the prime future effort in our program for Ft. Detrick and will dominate year two so far as <u>interaction with the working environment</u> is concerned.

9.2 Hardware Design and Fabrication

Preliminary investigations that were successfully conducted with the TC217 video "chip" will be translated in year 2 into a full-fledged design and fabrication effort of a preliminary prototype of PCM (to be followed in year 3 by the fabrication of several of these prototypes). The year 3 effort will be done in conjunction with Loral (San Jose, CA) who are the prime contractors of the major military hospital effort in imaging (called MDIS - Medical Diagnostic Imaging System). MDIS is primarily concerned with radiological imaging and is installed at major military medical hospitals throughout the USA with installations planned in Hawaii and South Korea.

Loral has agreed to work with Kensal staff in providing hardware (if required) and consulting advice that will assist Kensal staff in designing PCM in such a way that it may be interfaced with MDIS during year 3. Since experienced video electronic engineers who are willing to work at a small business such as Kensal are hard to find in Arizona, Kensal staff will contract out a great deal of this work to organizations in California where such talent is readily available.

Future plans, therefore, are to have Kline Research (Reseda) build the video imaging system (both CCD and LSDA). Ken Crocker Consulting (San Diego) will build two interface cards to the Apple Macintosh host (9500/120) for the purposes of (1) moving the mechanical system that controls the position of the microscope slide and (2) "captures" the video produced by the Kline Research cameras. After investigating possible collaborators in Arizona, we , again, have elected to go to California (as recommended by Boeckeler Instruments, Inc.) and contract with Optical Systems Corp. (Valencia) for the mechanics and optics of PCM. This work was initiated in the fourth quarter of year 1 and, in the first quarter of year 2, it is expected that a design will be completed followed by construction of a preliminary protype of PCM during the second and third quarters of year 2 and production of several additional units during year 3. This will be reported in our second Annual Report.

9.3 Software Research and Development

This effort is behind schedule. It was initially decided that Kensal staff (consisting of several software programmers) would produce a Coding Standard and then a Software Specification for the effort involved in producing the software for PCM. After two months effort in the third quarter of year 1, it was recognized that programmers cannot write coding standards and that a full-fledged software engineer was needed. Since it is almost impossible to recruit senior people into a company that is incrementally funded (such as with this grant) we outsourced this work to PlanetTools (Carefree, AZ). Jay Nance (head of PlanetTools) thus started an effort in the third quarter of year 1. Completion is scheduled for the first quarter of year 2. He will produce a software specification and has already written a coding standard. Again, this is a departure from previous plans due to the circumstances described above. Results so far are excellent and it is expected that in year 2 all software will have been completed with preliminary testing taking place on the preliminary prototyped being built in California. In year 3 the amount of effort on software will decrease as only modifications are made and debugging is completed. This will allow us, during year 3, to build and install the additional prototypes. The number of prototypes will depend upon the pricing obtained from our subgrantees.

9.4 Rapid Prototyping

The prototype that ARPA requested be built in a rapid prototyping mode was, as planned, contracted to Boeckeler Instruments, Inc. (Tucson, AZ). The instrument is complete and is being debugged. (All failures have been due to vendor-purchased components and not to any deficiency introduced by Boeckeler Instruments). It is completely described in Section 3 and is performing to its specifications. The only problem has been that, due to malfunctions of the Kodak driver board, the lensless image occasionally is missing a large number of lines. This is currently being corrected as the problem appears to be a "ground loop." Also, one of the "frame grabber" cards from Matrox has developed an input problem where it does not digitize the incoming video. Matrox will replace this card. It is expected that during the first quarter of year 2 the instruments from Boeckeler will be accepted and placed at the hospital of Luke Air Force Base and at the Mayo Clinic. At that time a two-month telemedicine experiment will be conducted during which approximately 50 microscope slides, selected by mutual agreement between Luke and Mayo, will be analyzed by telemedicine. After that, further demonstrations are planned, but the locations are still under discussion. Several organizations have expressed interest in lensless scanning and would like to adopt the Boeckeler instrument to their needs. Other locations (both military and

civilian) have also asked for demonstrations. This list includes professional societies such as the College of American Pathologists.

9.5 Negative Results

As required in Annual Reports under this grant, we herewith present negative results of some significance. There are two such results.

First, Boeckeler Instruments, due to failure of vendor-purchased instruments and hardware, has been unable to deliver their prototypes as rapidly as ARPA initially expected. As noted above, this problem is being rapidly solved. It is not fundamental. The vendors involved in delivering unsatisfactory instrumentation are replacing same at their earliest convenience.

Second, it might be considered a "negative result" that Kensal staff has proved inadequate to (1) produce a software specification for PCM, (2) has been found to have insufficient skills to generate the high-definition television portion of PCM, and (3) requires that the major optomechanical assembly be produced by others. This has slowed our progress but, in the opinion of the PI (Principal Investigator), will, if anything, result in a more superior instrument than one built entirely inhouse. Whereas it had been expected to produce a preliminary of PCM during year 1, this milestone will not be reached until sometime in the third quarter of year 2. This is not a serious negative result in that, in the meantime, we have the rapid prototypes produced by Boeckeler which performed <u>exactly</u> as will the PCM with the exception that they are far more extensive and require far more space on the desktop than would PCM when completed.

In conclusion we feel that the program is going well and that, except for minor delays, is producing excellent results.

9.6 Proposed Staff and Budget Changes

Since research and development is a dynamic process, it is not surprising that a proposal submitted in March of 1994 no longer is applicable exactly to the interval 1 October 1995 to 30 September 1996. Thus this section outlines a rebudgeting of funds for this same interval.

9.6.1 Staffing

Outsourcing has already been discussed above. It has led to the changes described below. Once the project started certain consultants and subcontractors were replaced by more suitable candidates as follows:

Original Subgrantee	Replacement Subgrantee
Neuman DeBell	John Sparks and staff of Optical Systems Corp.
Loral/Lockheed	Loral (Lockheed used only in year 1)
Motorola	Nance, Guerin, Kline (See Sections 7 and 8)
Boeckeler	No change (But used only in year 1)

9.6.2 Budgeting

The amount of the original budget for year 2 is adequate, but funds need to be redeployed to reflect changes in staffing. In particular it is recommended that funds allocated to Loral for a

workstation should now be moved to Optical Systems Corp. in that they will be fabricating the preliminary prototype of PCM. Also the Apple Macintosh 650 that was proposed for use by Loral in 1994 as part of their workstation is now obsolete and will be replaced by one of the far more modern Macintosh 9500/120's that have already been procured by Kensal. Loral will still be used for consulting as they will play a major role in year 3. The original budget and proposed budget, given below, reflect the above changes.

		Year 2 Budget	
		Original (approved 9/94)	Proposed (10/95)
1.	SALARIES (W-2 and 1099) [1]	160,000	160,000
2.	BENEFITS [1]	none	none
3.	CONSULTANTS [2]		
	Neumann	20,000	none
	DeBell	7,000	none
	Nance [2]	none	27,000
4.	EQUIPMENT		
	PCM Assemblies [3]	12,931	12,931
	Optics [3]	21,653	21,653
	Workstations [3]	54,220	54,220
5.	SUPPLIES & MATERIALS [3]	11,281	11,281
6.	TRAVEL		
	PHX-SFO	1,832	1,832
	PHX-SAN	1,408	1,408
7.	ADMINISTRATIVE SUPPORT	none	none
8.	INDIRECT COSTS	62,400	62,400
9.	MISCELLANEOUS [4]		
	Loral	76,300	26,300
	Optical Systems Corp.	none	50,000
10.	TOTAL COST	429,025	429,025

Notes:

[1] Kensal in some cases now supports medical insurance for certain of its employees. When this is done, costs will be taken from salaries.

[2] Hans Neumann and Gary DeBell are not participating in research under this grant since their work on mechanics and optics, respectively, is being done under contract to Optical Systems Corporation. However, due to the increased complexity of our software effort, an outside consultant, J. Nance, will be required to code software for PCM (PC Microscope).

[3] Equipment and supplies needed for the first PCM workstation will be procured by Kensal and furnished as required to Optical Systems Corp.

[4] Optical Systems Corp. will design, fabricate, and test the first prototype PCM workstation. This workstation will be designed to be interfaced to the Loral MDIS system. Procurement of Macintosh computers and color displays from Loral will not be necessary. These will be furnished using existing equipment at Kensal Corp.

i. Dr. Madjidi was did not clarify how the diagnosis was understood by the computer but upon suggestion of AI, she agreed that this must be the way SNOMED codes were generated from the logged diagnosis.

ii. Literature from HBO&Company, 301 Perimeter Center North, Atlanta, Georgia 30346, Phone (404) 393-6000. Descriptions from "STAR SOLUTIONS" leaflet (HBOC 8-794).

iii. RAID - Redundant Array of Inexpensive Disks

iv. Apple has successfully accomplished image stitching of a mosaic of pictures taken with an ordinary lensed camera in their QuickTime VR extension.

v. See my book summary, "Management Information Systems, A Managerial End-User Perspective", KSCTR-9506, p. 7.

vi. It is also analogous to the way we build consensus with our note-taking effort for greater accuracy and information content.

vii. *Decision support systems* (DSS) are interactive, computer-based information systems that use decision models and specialized databases to assist the decision-making processes of managerial end users. A DSS provides ad-hoc report generation, analytical modeling, data retrieval, and information presentation capabilities. Thus DSS's differ from the pre-specified responses generated by *information reporting systems* (IRS) which provide information products that support data-to-day decision-making. *Executive information systems* (EIS) are tailored to the strategic information needs of top or middle management. EIS provides the current status and projected trends for key factors selected by top executives.

viii. This eliminates communication traffic problems between nodes. Data being transmitted is first stored and waits until bandwidth is available for transmission. This does two things, 1) frees up the end-user's terminal to work on other things while the data is queued and, 2) ensures fault-tolerant delivery of data incase of temporary disconection during transmission--data can be resent.

ix. Medical Reords Institue Web Pages (http://www.medrecinst.com/)

x. Many hospitals still keep much of the patient record on paper in a folder labeled with the patient's unique HIS index number.

xi. Medical Records Institute, 567 Walnut Street, P.O. Box 289, Newton, MA 02160 USA; Phone: (617) 964-3923

xii. See Appendix A, <u>Networking And Standards</u>, paragraph A.2 and the ATM Forum, at WWW URL http://www.atmforum.com

xiii. The following is from the Medical Records Institue, "Standards in Health Care Informatics" web page (http://www.medrecinst.com/).

xiv. Medical Records Institue, 567 Walnut Street, P.O. Box 289, Newton, MA 02160 USA, (617) 964-3923

xv. This directory costs $42.50

xvi. Much of the following comes directly from a Los Alamos National Laboratory web page (http://www.acl.lanl.gov:80/sunrise/).

xvii. *Message transformation* is translating and mapping data to other formats, splitting (sending to multiple destinations) or compounding (receiving from multiple sources) messages.

xviii. According to a GartnerGroup Research Note (April 13, 1995; HCV), larger hospitals choose a "best-of-breed" approach to selecting applications (i.e., the best lab system, the best pharmacy system) with little concern for each application's overall architectural fit.

xix. EDI - Electronic Data Interchange

xx. ASTM - American Society for Testing and Materials

xxi. SAIC Headquarters, 10260 Campus Point Drive, San Diego, CA 92121, (619) 546-6000

xxii. J.R.Beyster is currently SAIC's Cheif Executive Officer

xxiii. Statement from SAIC 1995 Annual Report, Information Technology, located at WWW URL: **http://139.121.25.30/**

xxiv. The following information is from MEDSITE URL: **http://bender.brooks.af.mil/www/chcs.html**

xxv. After Action Report by Lt. Col. Lynn Ray, available through MEDSITE WWW URL: **http://bender.brooks.af.mil/**

xxvi. NAVHOS - Naval Hospital

xxvii. According to Ms. Monica Brown, HBOC Investor Relations, (404) 668-5926

xxviii. Cost dependant on what client buys--typical HIS is 4 to 5 modules.

xxix. See section 1.5 Computer-based Point of Care

xxx. From Charles Spurgeon, WWW URL: http://wwwhost.ots.utexas.edu/ethernet/descript-10quickref.html

xxxi. Obtained from WWW URL: http://www.atmforum.com/

APPENDIX B

1996 ANNUAL REPORT FOR DAMD17-94-J-4500

TABLE OF CONTENTS

1. INTRODUCTION

This is the Fiscal Year 1996 Annual Report for grant DAMD-94-J-4500 (Dual-Use Telemedicine Support System for Pathology) for USAMRMC (U.S. Army Medical Research and Materiel Command). The report covers the use and application of a novel workstation for pathology that integrates both lensless and lensed imaging of surgical pathology specimens mounted on standard microscopy slides. The research reported here is being conducted in parallel with National Institutes of Health grant 5 R44 GM44420-03 entitled "Image Handling System for Pathology and Telepathology." The research involves the use of two workstations built under the USAMRMC grant in a telepathology experiment where one workstation is currently operating at the Mayo Clinic (Scottsdale, Arizona) and the other at Luke Air Force Base (Litchfield Park, Arizona). This project has been under the direction of Lane Garrett of Scottsdale who has served both as a employee of the Kensal Corporation and as an independent consultant. This research has involved not only Mr. Garrett but also Drs. Louis Weiland, Kevin Leslee, Hermilando Payen, and Felix Mamani. These pathologists have produced a protocol for the experiment and are carrying out a double-blind study that will compare the performance of telepathology with ordinary manual microscopic pathology for a selected group of about 50 microscope slides. A report on this study will be made in our final report upon completion of the research.

As regards research, finding ourselves will a few hundred digital images (both lensless and lensed) of surgical pathology specimens, the Kensal staff has been investigating methodology for generating not only a database but also a "Virtual Microscope" that is simply an elaborate digital recording on CD-ROM that can be employed by the computer user to simulate use of the L/L Microscope (Lensed/Lensless Microscope). Copies of this CD-ROM have been distributed to several interested parties and, of course, the Advanced Research Projects Agency.

This FY 1996 Annual Report is divided into five sections as follows:

- Section 1 (this section)
- Section 2 - This section gives both the protocol for the double-blind study commenced between Mayo Clinic and Luke Air Force Base in early calendar 1996. Because of malfunctions of the workstation at Luke Air Force Base, the exact completion date is uncertain and is now estimated as taking place in the first calendar quarter of 1997.
- Section 3 - In order to design the CD-ROM format called the "Virtual Microscope" an extensive survey was conducted of other CD-ROM databases of surgical pathology images developed both by government laboratories, academic institutions, and grantees such as our corporation.
- Section 4 - Since the eventual interface between the workstations and the hospital will be with a HIS (Hospital Information System), HIS's developed nationwide were surveyed. This section documents our findings. Note that in the first annual report similar documentation was provided for pathology information systems.
- Section 5 - A major event occurred during FY 96 that will impact all of our future research. This was the successful culmination of research under National Science Foundation grant DMI-9460231 (Research on Lensless Microscopy). As of September 1996, Kensal had in its laboratory the first new high resolution lensless scans using the state-of-the-art fiber optics combined with the latest high-resolution silicon-based linear detector arrays. As of only a short time ago we were able to digitize full coverslip images at 20 thousand picture points per square millimeter. This compares with a pixel density of only 6 thousand pixels per square millimeter in the current workstations. This, we believe, now solves the problem that lensless images appeared "fuzzy" to some pathologists to the extent that they were unable to identify regions of interest in those scans. This remarkable achievement under the NSF grant makes all current

workstation obsolete. We are therefore suggesting that funds available for FY 97 be reprogrammed, not to build more workstations but to retrofit the existing workstations. Therefore this section presents a new budget and justification thereof.

2. LUKE / MAYO 1996 TELEPATHOLOGY STUDY

The telepathology study between the Luke Air Force Base and the Mayo Clinic commenced in early 1996 and is continuing.

2.1 Protocol

The proposed double-blind telepathology study will use a minimum of 25 cases from the US Air Force Hospital at Luke AFB, Litchfield, AZ and a minimum of 25 cases from the Mayo Clinic, Scottsdale, AZ. The cases will include a broad sampling across organ groups. (No blood smears or Pap smears are to be included.) A cardboard template will be provided that shows the current guide image boundaries of the workstation. Microscope slides that have tissue outside this "view area" should not be used in the study. A separate protocol is being prepared by Kensal to describe recording data in a database. Kensal will be responsible for the compilation and analysis of the data and the final written report.

1. In the first step, Luke AFB will select a minimum of 25 pathology cases that that have been previously diagnosed and represent a broad but typical cross section of work at Luke. Each case and slide will be given a unique six digit number starting with "1" (signifying Luke) as the most significant digit. Traceability to the accession number will be held only at Luke in a separate log by Maj. Cooper as the cognizant individual. The new case and slide numbers and other pertinent data (to be determined) will be placed in a spread sheet format by staff of the Kensal Corporation. The second and third most significant digits "xx" will sequentially keep track of each case. A case may have multiple slides with consecutive numbers using the two least significant digits "yy". The fourth position "z" is presently reserved for future use. For example, 1xxz01 will indicate the first slide in case "xx".

   Initially, Luke will play the role of the "remote user" and take guide image(s) that will be transmitted to Mayo. Hi-mag images will then be taken by Luke as requested by Mayo. The reviewing pathologist(s) at Mayo may request additional hi-mag images after examining the initial image(s). Additional information may be requested and provided by Luke as appropriate. A number of iterations may take place for hard to diagnose cases.

   A mutually acceptable number of cases will be handled weekly to keep the work load at a reasonable level. Upon suitable notice, Kensal personnel will be made available to assist as needed.

2. Maj. Cooper, as the cognizant individual at Luke, will then assign new six digit numbers (with the most significant digit assigned a value of "2") for each case and slide used in step 1. The renumbered slides will be forwarded to Mayo for diagnosis using the TSS in "local mode", i.e., using the TSS as a self-contained instrument without any ISDN transmission.

3. In a third step, the renumbered slides will be analyzed at Mayo in the normal visual manner, i.e., without using the TSS. If time permits and staff is available, the same glass slides will be reviewed by a "panel of experts" at Mayo using normal visual diagnosis. If there is any disagreement with a diagnosis from steps 2 and 3, the reason(s) are to be ascertained, if possible, for the purpose of finding how the methodology can be improved.

4. In a like manner Mayo Clinic will play the role of the remote user and go through the above steps 1 through 3 with 25 or more of their cases. Six digit numbers for each case will be assigned by Dr. Weiland beginning with "3" as the most significant digit. Luke will then perform the role of "expert" pathologist. New numbers beginning with a "4" as the most significant digit will be issued when the actual glass slides (chosen by Mayo) are sent to Luke for both TSS local mode (step 2) and normal (step 3) analysis. For step 3, since there are insufficient pathologists to form a "panel," perhaps, AFIP would consent to participate by forming a panel.

2.2 Problems with Workstations Installed at Luke AFB and Mayo Clinic

- Work is required on the enhancement of contrast and color fidelity of the guide images to permit satisfactory selection of regions-of-interest (ROIs) for higher magnification images.
- There is an intermittent problem of the Video Relay sticking in the camera mode stage.
- The positioning of numerous guide image scans varies over time.
- The guide images deteriorate until they become unusable. Both the new and old software have been tested with the same results.
- Serious low amplitude oscillations develop which can only be stopped by "damping" the stage with slight hand pressure on the edge of the stage.
- Various amounts of both horizontal purple and yellow lines and vertical lines of discoloration appear usually several tenths of a mm wide.
- The focus control needs work. As it now is programmed, the operator frequently overshoots the focus point and has a hard time coming back, especially at 20x and 40x.
- There is much difficulty navigating when in Windows NT.
- The monitor screen tends to drift slightly in size and position requiring realignment with the touch screen grid.
- When "Quit" is accidentally activated, the whole session aborts.
- The stage is unstable when jumping to high-magnification positions.
- When in Stand-alone Mode and the Guide image is scanned and brought to the screen, there is no provision for recording a message with the Guide image.
- In the Dual Mode of operation, the sending station can add comments to the Guide image file and receive back any comments that the receiving (expert) station has made on the high-magnification images; however, the comments are apparently not recorded with the high-magnification images that are saved at the expert station.
- In the Single Station Mode, it is possible to jump around the viewed locality when in the "Get Hi-Mag" function. A problem arises in that there appears to be a finite number of times that this can be done before "crashing" and losing the whole session on that case.
- When viewing the next high-magnification images, the doctor may start dictating shortly after it opens. The image number, magnification and position are updated very slowly sometimes causing the doctor to pause for the update.
- After a period of multiple scans, the whole guide image shifts left and does not correct itself.
- In the "Local Mode" the system has a limitation of approximately 16 "jumps" before it crashes.
- Inability to go back and review high magnification images that have been saved and record additional comments is a problem.
- The guide images lack some contrast unless the slides are very good initially.

- Some kind of backup capabilities should be built into the TSS1 software and automated.
- The Mayo workstation is consistently showing a warp between rows K & J with displacement to the left.
- The Mayo system exhibited significant height distortion.
- When in the "Local Mode" it is easy to forget to record the Guide image. In the "Dual Station Mode" the opposite problem can occur where the guide image can get saved twice.
- It would be useful to be able to selectively erase the green squares on the guide image that show the ROIs (Regions of Interest) such that the guide could be rescued without clutter.
- Occasionally memory errors occur for no apparent reason.
- The current sound cards are unacceptable for any production work.
- Still experiencing problems when recording audio for a given image. If the operator does not hit the stop or half button just right, the record window jumps behind the TSS1 window and keeps the recorder running. The TSS1 program assumes that recording has stopped and allows the operator to go on to the next image. A second recording window will then come up when the operator wishes to record but will only function for about one section then it seems to lock up. The only recourse is to abort the whole session, go to Windows NT where the first window shows the recording continuing and stop the recording.
- Problems occur when transmitting requested high-magnification images from Luke to Mayo. The error messages "Unable to transmit request file" and "Memory could not be written" occur several times. High-magnification images are lost.
- Unable to play back the original history and comments without having to reload the original guide image when receiving previously requested high-magnification images back from the transmitting site.
- Coordinates on the guide image are off according to the actual high-magnification being observed.
- The damping coefficients or stage characteristics seem to change with time and/or usage.
- When trying to do the first downloads to the SyQuest drive were unsuccessful, patterns were observed that sometimes plaque the scans of the guide images. This indicates that the noise dots are related to the scan sync and/or the phase-locked-loop.
- Operating in the "Local Mode" and starting to do a diagnosis going through requested high-magnification images from 2/11 through 11/11, the system jumped back to image 2 when going from high-magnification image 9 to 11.
- Two pixels are dead on the Luke LDA.
- After a telepathology session, it is hard to know what was actually completed successfully since there is no status information or receipt of document information.
- The 2 MB hard drive fan on the Luke workstation has developed a noise and needs to be replaced.

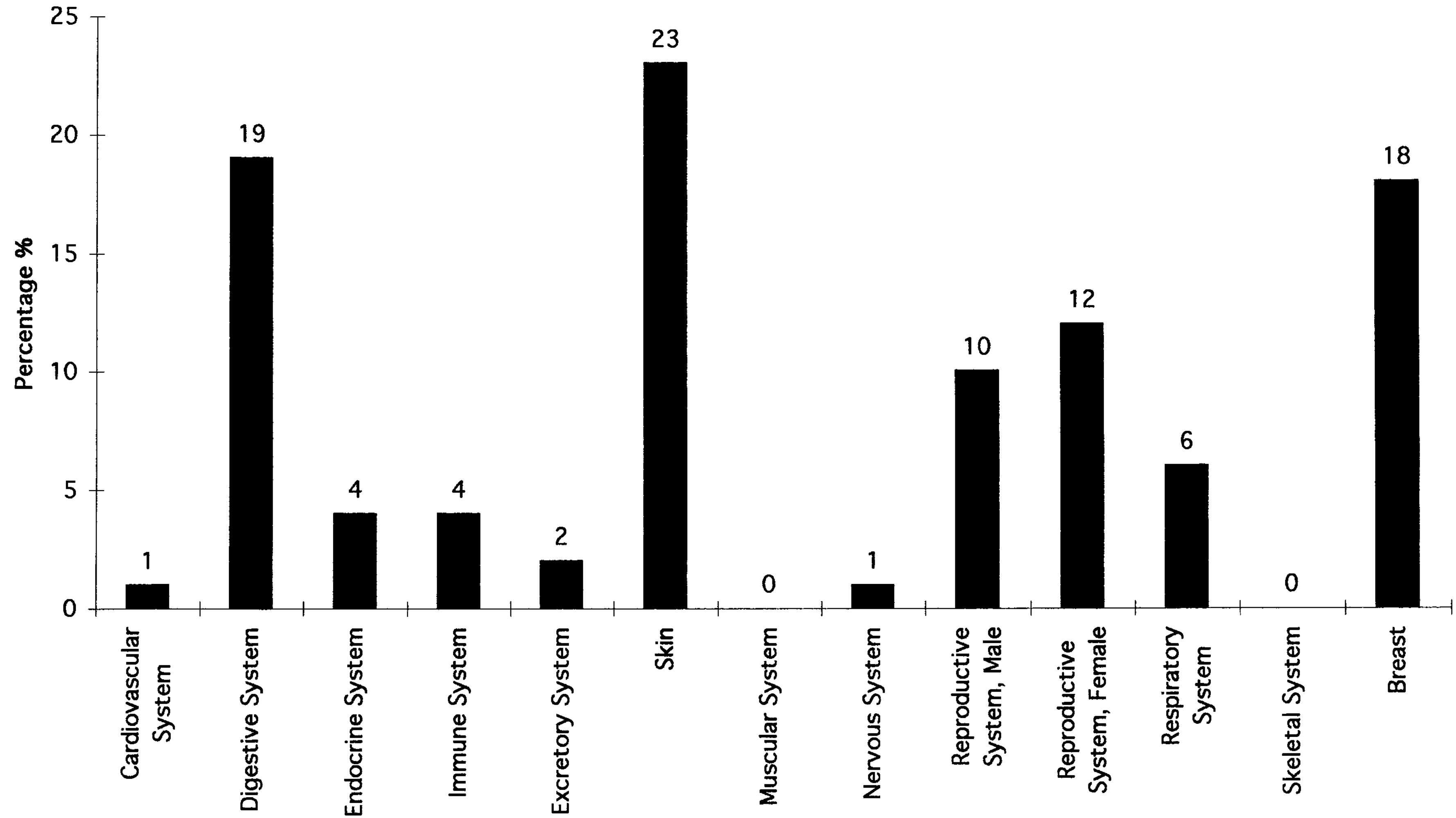
2.3 Tabulation of Microscope Slides Under Study

Table 1 contains information regarding the Luke Air Force Base / Mayo Clinic Telepathology Experiment including slide history and status as of October 15, 1996. This data has been added to the organ systems histogram (Chart 1) for slides done only between Kensal and the Mayo Clinic prior to the installation at Luke AFB that occurred in mid-1996.

Table 1 - Luke/Mayo Telepathology Slide History and Status; 12/10/98

Mayo Cases	Luke Cases	Local Mode Done	Sent To Kensal	Cleared Out	New #	Dual-mode Im. Sent	Hi Mags Requested	Hi Mags Sent	Dual Mode Diagnosis	Sent to Kensal	Cleared Out	Comments
		Dr. Weiland			Dr. Weiland to Luke		Dr. Payen					
501000		7/16/96	7/24/91	8/16/96	501000	8/16/96	33838					Requests
502000		7/16/96	7/24/91	8/16/96	502000	8/16/96	33838					8/27/96
503100		7/16/96	7/24/91	8/16/96	503100	8/16/96	33838					Requests
504100		7/16/96	7/24/91	8/16/96	504100	8/16/96	33838					Requests
505000		7/17/96	7/24/91	8/16/96	505000	8/16/96	33838					Requests
506000		7/17/96	7/24/91	8/16/96	506000	8/16/96	33838					Requests
507000		7/17/96	7/24/91	8/16/96	507000	8/16/96	33838					Requests
508000		7/17/96	7/24/91	8/16/96	508000	8/16/96	33838					Requests
509000		7/17/96	7/24/91	8/16/96	509000	8/16/96	33838					Requests
510000		7/17/96	7/24/91	8/16/96	510000	8/16/96	33838					Requests
		Dr. Weiland			Dr. Leslee to Luke							
511100		8/26/96	8/27/96	8/26/96	511100	8/28/96						
512100		8/26/96	8/27/96	8/26/96	512100	8/28/96						
513000		8/26/96	8/27/96	8/26/96	513100	8/28/96						
514000		8/26/96	8/27/96	8/26/96	514100	8/28/96						
515000		8/26/96	8/27/96	8/26/96	515100	8/28/96						
516000		8/26/96	8/27/96	8/26/96	516100	8/28/96						
517000		8/26/96	8/27/96	8/26/96	517100	8/28/96						
518000		8/26/96	8/27/96	8/26/96	518100	8/28/96						
519000		8/26/96	8/27/96	8/26/96	519100	8/28/96						
520000		8/26/96	8/27/96	8/26/96	520100	8/28/96						
	Drs. Payen, Mamani				Dr. Payen to Mayo		Dr. Weiland	Dr. Mamani	Dr. Weiland			
	601000	7/30/96	8/7/96		601000	8/16/96	8/16/96	33834	8/23/96			Lost some
	602000	7/30/96	8/7/96		602000	8/16/96	8/16/96	33834				Re-Request
	603000	7/30/96	8/7/96		603000	8/16/96	8/16/96	33834				Lost some
	604000	7/30/96	8/7/96		604000	8/16/96	8/16/96	33834				Re-Request
	605000	8/7/96	8/7/96		605000	8/16/96	8/16/96	33834				Lost some
	606000	8/7/96	8/7/96		606000	8/16/96	8/16/96	33834	8/23/96			Re-Request
	607100	8/7/96	8/7/96		607100	8/16/96	8/16/96	33834				Lost some
	608000	8/7/96	8/7/96		608000	8/16/96	8/16/96	33834				Re-Request
	609000	8/7/96	8/7/96		609000	8/16/96	8/16/96	33834				Lost some
	610000	8/7/96	8/7/96		610000	8/16/96	8/16/96	33834				Re-Request
		Dr. Mamani			Dr. Payen to Mayo		Dr. Leslee					
	611000	8/19/96	8/23/96		611000	8/16/96	9/5/96					
	612000	8/19/96	8/23/96		612000	8/16/96	9/5/96					
	613000	8/19/96	8/23/96		613000	8/16/96	9/5/96					
	614000	8/19/96	8/23/96		614000	8/13/96	9/5/96					
	615000	8/19/96	8/23/96		615000	8/13/96	9/5/96					
	616000	8/19/96	8/23/96		616000	8/13/96	9/5/96					
	617000	8/19/96	8/23/96		617000	8/13/96	9/5/96					

Chart 1 - Organ Systems Histogram



3. A REVIEW OF DIGITAL IMAGE DATABASES FOR PATHOLOGY

This section is the text of a review paper that is being prepared for submission to the College of American Pathologists.

During the past decade, physicians have witnessed dramatic advances in computer technology. The rapid development of video and computer based communications of medical information has now made it possible for a physician to examine a patient in a remote location. The support for digitized examinations in radiology is quite extensive, however, this kind of support has been lacking in the field of pathology. In fact, the dynamics of gross and microscopic pathological exams have remained basically unchanged in the last century. Pathologists when looking through the eyepieces of their microscopes, still find inverted images of only a small portion of the specimens to be analyzed.

Modern laboratories have indeed begun to utilize computers, but mainly for the management of text. Today telepathology systems do exist which afford the means to transfer pathological reports and provide verbal communication links between pathologists. Yet the image more than the written word drives diagnostic decisions. The truth is that very few systems have the capability of handling images. In order to bring pathology into this era of high technology, images must be created and stored in a digital format.

Many significant benefits could evolve from the use of digitized microscopic images. These images may be used to aid in diagnostic efforts, to serve as educational resources, and to facilitate communication between pathologists. Being able to view an image over a communications network would reduce referral time from a remote pathologist by eliminating the need to transfer the glass slide. Images organized to form a digitized atlas of surgical pathology could serve to lessen the time a pathologist spends consulting reference books and other pathologists. Medical students would also benefit greatly from computer-assisted training technology, especially considering the vast accumulation today in digital image databases.

3.1 Current Status of Telepathology

A problem exists with ordinary microscopy in that a pathologist is only able to see a minuscule portion of the coverslip even when using the lowest power objective lens (2x). The Kensal Corporation is currently working to resolve this problem through the development of a telepathology workstation which will integrate Kendall Preston, Jr.'s patented "lensless microscopy" with lensed full-color imaging (U.S. Patent 4,777,525). "Lensless microscopy" enables a pathologist to capture and display on screen an image of the *entire* coverslip of a glass slide. This image is referred to and essentially used as a "guide image." The workstation is unique in that it can register and integrate each full-coverslip scan with the high magnification images taken from the corresponding coverslip. Coordinates of each high magnification are automatically recorded. This enables the pathologist to return to the high magnification images at the exact locations requested.

In 1994, the Defense Advanced Research Project Agency (DARPA) granted monies to Kensal Corporation to build and test telepathology workstations that integrate lensless full-coverslip scanning with lensed imaging. In 1996, telepathology field trials were conducted in Arizona with these workstations between Mayo Clinic of Scottsdale and Luke Air Force Base in Litchfield Park. Guide images were produced at the "host" station and sent to the referring pathologist at the second station over ISDN (Integrated Services Digital Network) lines. The referring pathologist used this image as a "global reference" to select regions of interest to be magnified. Once this was completed, the magnified images were sent back to the host pathologist and high magnification images were produced with a Sony color camera mounted on top of a

Nikon microscope. The high magnification images were transferred back to the referring pathologist and one of two paths was selected: either a diagnosis was rendered or the remote pathologist requested more high magnification images. Dictation and annotation could be included while viewing each image. Correct diagnoses were made for almost every tissue sample tested.

The telepathology workstation, now formally called the TSS1, organizes the images and sound files in a digital image database used to record, store, and manage the results of each examination. From the field trials, Kensal Corporation has produced a collection of images, including both the full-coverslip images from the lensless scanner and the high magnification images produced from the lensed, high magnification camera. This data set is unique because it utilizes two registered imaging technologies. To date, Kensal has accumulated a library of approximately 300 images.

3.2 Other Collections of Pathology Images

The production of this unique database and imaging system has led Kensal to look at other databases available to pathologists and medical students to assess their function and compare how different areas are covered. Several other institutions, ranging from academic to government, have begun handling images in digital formats, forming atlases and databases along the way that can be used as references for pathologists and as educational tools for medical students and residents. Whether in the form of an optical disk or a file on the Internet, we have found this information to be useful and easy to access.

The following is an illustrative review of some digital image databases available for pathology. This selection of CD-ROMs represents some of the most comprehensive collections of digitized images of microscopic tissue available. There are a variety of other CD-ROMs on the market as well which serve as electronic references for other pathological disorders, including hematology, cytology, gynecological, ophthalmic, and orthopedic pathology, however we were unable to include all of them in this review.

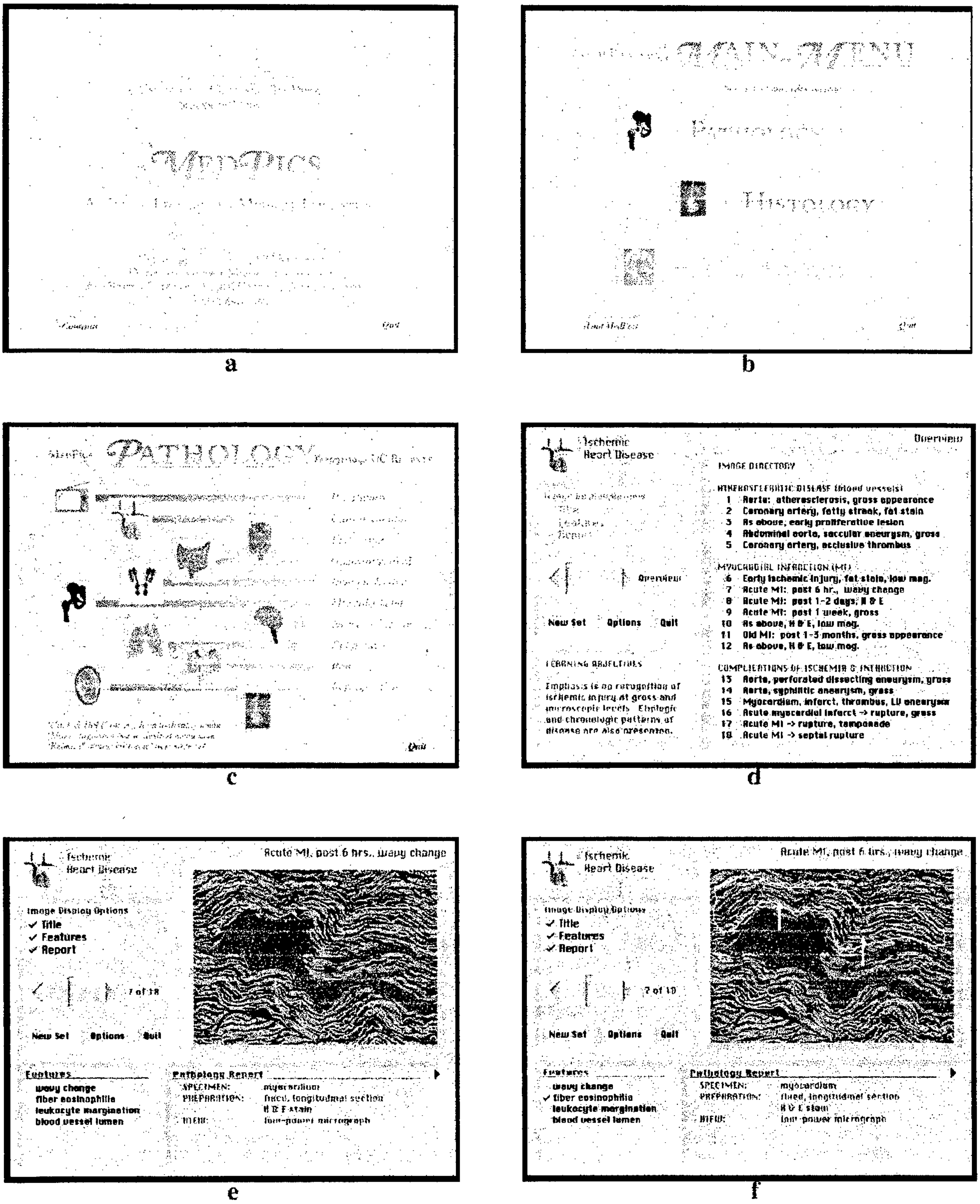
The six image databases profiled were chosen from four different areas: Academic (University of California at San Diego and the University of Utah), Government (the Armed Forces Institute of Pathology), commercial (Chapman & Hall and Mosby Multimedia), and military subcontract (Kensal Corporation for the U.S. Army Medical Research and Materiel Command).

3.3 Educational Image Databases

Many schools are now beginning to realize what a powerful educational tool image databases can be for resident and medical student training. The University of California, San Diego has developed a database called MedPics that is now commercially available from Micron BioSystems (Denver). MedPics is a computer-based image delivery system with supporting text fields and on-screen graphics to assist in lessons on normal and abnormal structures and functions of the organs. It has been used as an integral part of the Human Disease Course at UCSD since 1992.

Currently, MedPics contains just over 600 images, including gross and microscopic examinations, x-rays, diagrams, and electron micrographs. You may choose to view images from ten different organ systems catalogued under either "Pathology" or "Histology" (Fig. 1-b). The section titled "Hematology" is expected to appear in next year's Version 2.0.

Viewing a slide set is as easy as clicking on the system's icon (Fig. 1-c) which takes you to a brief overview of the images available in that category (Fig. 1-d). You can then page through the images which are accompanied by an informative title, a list of identifying features, as well as a pathological report. The report typically includes information on the specimen such as the



a b c d e f

Fig. 1 - Major Frames from MedPics CD-ROM.

preparation and staining, the magnification used, any pathological findings, clinical pathologic correlation, pathology pointers and lastly the credits. One special feature for students is the Quiz Mode, which displays the images without giving any other information until it is requested. This program has had a positive impact on the second year medical students and has also improved student attitudes toward computer-based resources and development.

Some schools have developed larger archives of digital images. The University of Utah has an archive of over 1800 images in their electronic laboratory WebPath, which are available both on a CD-ROM and via the World Wide Web. These images demonstrate gross and microscopic findings associated with human disease, and are used extensively in the medical student pathology course at the University. Diagrams, electron micrographs, and X-rays are also used, but to a much lesser extent than with MedPics. Each image is supported by a text field to assist in learning about normal and abnormal features for the particular disease displayed.

WebPath was produced by Dr. Edward Klatt two years ago to support his pathology courses for second year medical students. He explained that having the course materials on the Web avoided many labor intensive activities associated with maintaining sets of kodachromes and glass slides. He decided last spring to put the material on a CD-ROM for two reasons. First, the slowing down of the Internet became so bad that modem access for the images was frustrating. Second, foreign sites were encountering bandwidth problems when trying to access the images over the Internet. Both of these problems have been resolved with the availability of the CD-ROM.

Utah has six major resource categories that images may be viewed under: General Pathology (images by mechanisms of disease), Organ System Pathology, Laboratory Exercises, Mini-Tutorials, Clinical Pathology, and Histopathology. Recently, two more categories were added which are the Examination category that has over 1500 multiple choice questions and sample essay items for students to work with, and the WWW Medical Resource Sites, which contains several information resources that are related to the science of medicine (Fig. 2-a).

Viewing an image is again as easy as one, two, three. By clicking on one of the image categories of the main menu you will be shown a more comprehensive list of that category (Fig. 2-b). For example, if you click on Organ System Pathology, a list of all the available organ systems will come into view. From that list, you choose the specific section of images you would like to view, such as Dermatopathology Index, which will transfer you to an index of images for that section (Fig. 2-c). This index also states whether the images are gross or microscopic. By clicking on the image of your choice, a full color, text supported image will appear on the screen within a matter of seconds (Fig. 2-d). Finally, this screen gives you the option to move directly forward or backward to other images without returning to the index as well as to return to the index directly.

3.4 Government Databases

The Armed Forces Institute of Pathology (AFIP) serves to provide the military, Veterans Administration, and federal and civilian pathologists with consultations, education, and research services. The AFIP has recently taken advantage of the rapidly expanding Internet. They offer many new services, which are displayed on the opening page of the Web site (Fig. 3-a). Some of the choices on their home page include a description of their mission, listings of educational programs and other services sponsored by AFIP and ARP (the American Registry of Pathology), an on-line edition of The AFIP Newsletter, information from the National Museum of Health and Medicine, a sampling of the AFIP Atlas of Tumor Pathology now available on CD-ROM, as well as an option to reply with feedback.

Since 1993, the AFIP has offered a useful service for those pathologists who are able to digitize images. Through this telepathology program, pathologists can transmit images using a file

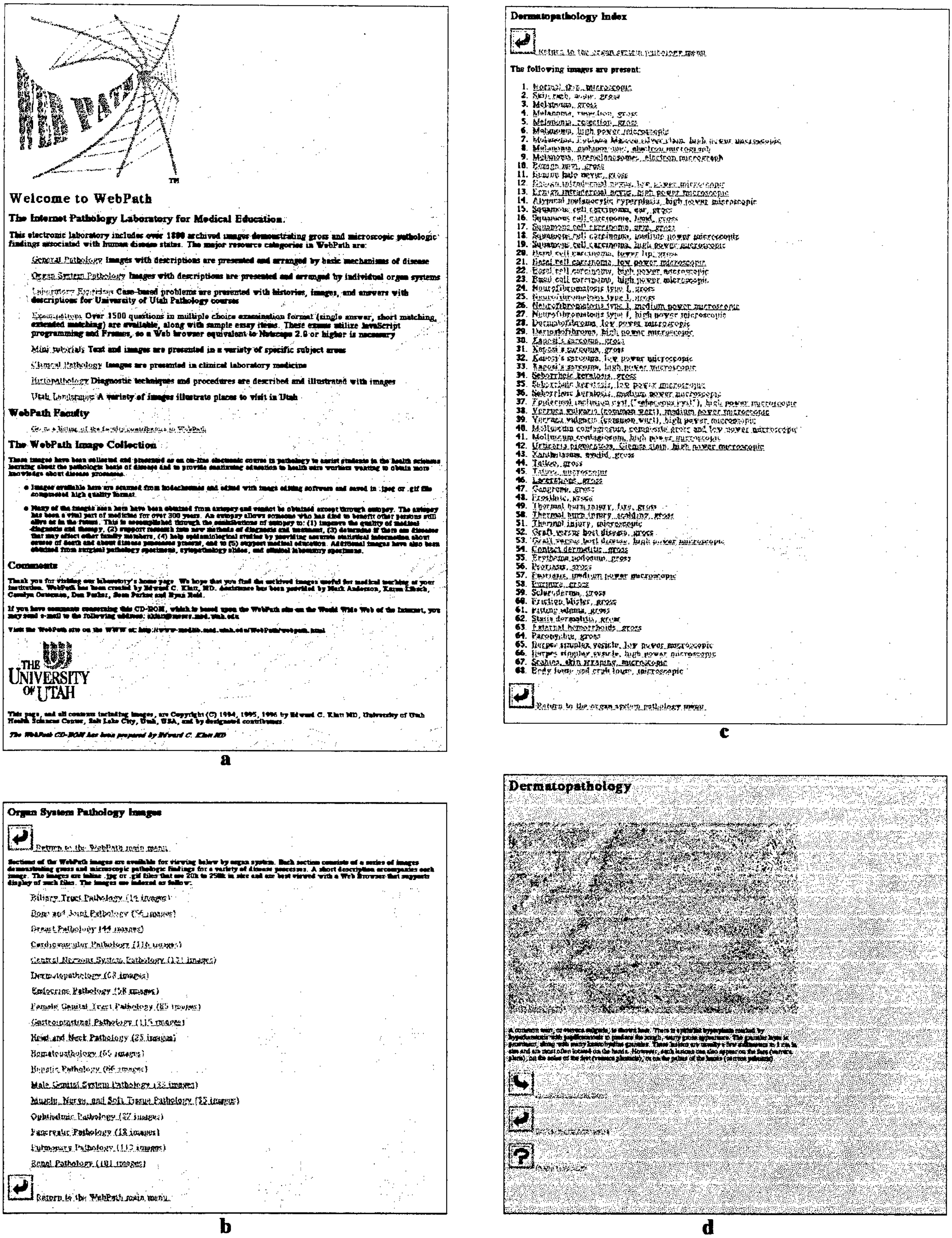


Fig. 2 - Major Frames from WebPath.

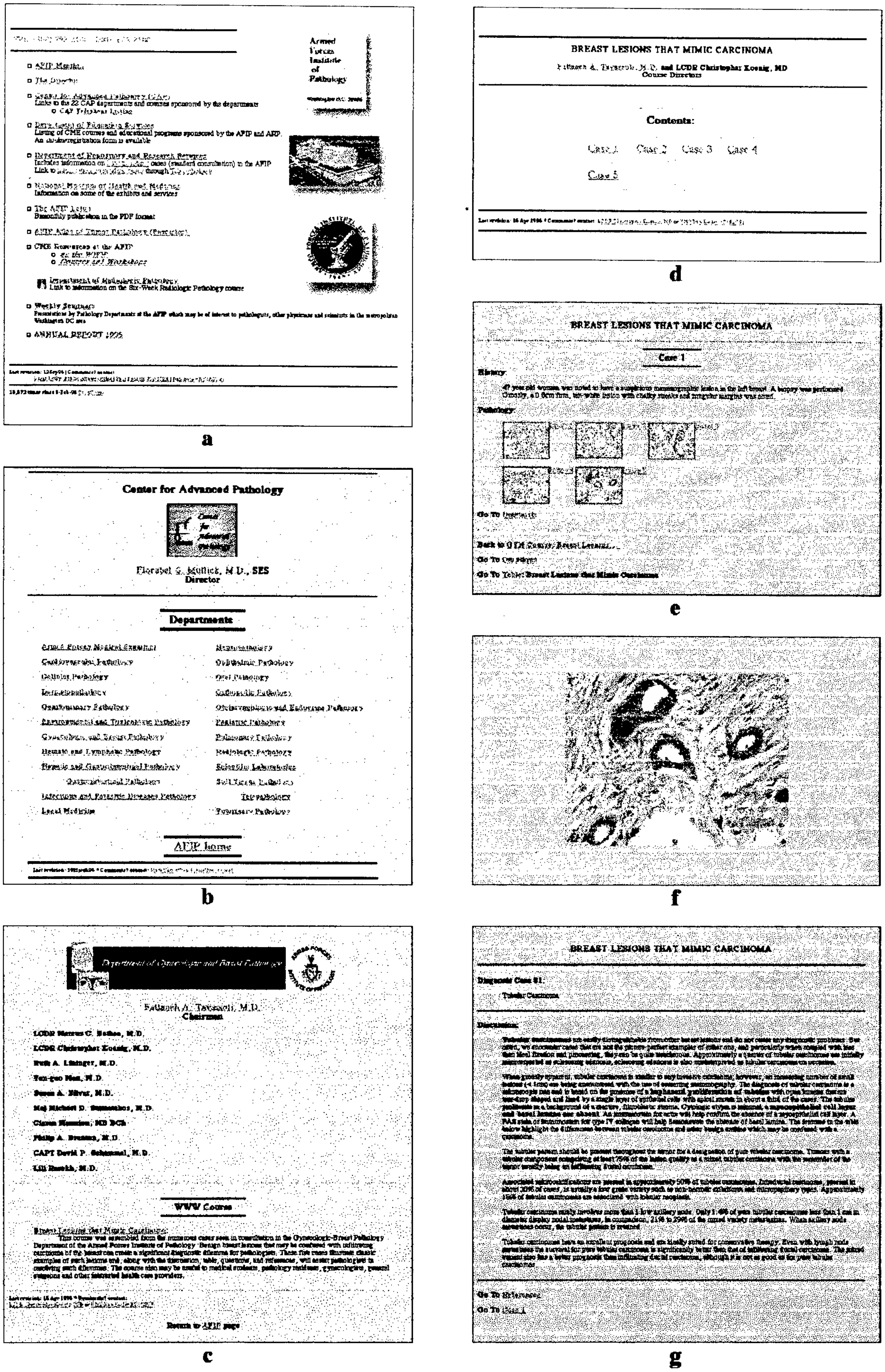


Fig. 3 - Major Frames from the AFIP Web Site.

transfer protocol (ftp) to the AFIP and receive rapid, expert diagnostic opinions. A report is faxed to the pathologist within four hours. The number of cases processed using telemedicine is still small compared to the 50,000 cases received each year by the AFIP, but the number is increasing.

Another service now available on the World Wide Web are the virtual courses developed by The Center for Advanced Pathology (CAP), a division of the AFIP. Some of the 22 sub-specialty departments of CAP (Fig. 3-b) offer courses which include case histories, gross and microscopic images, and discussions. The courses typically include a number of case studies (Fig. 3-d) which were once seen in consultation with the AFIP. Each case is displayed on a page which shows "thumbnails" of each image (Fig. 3-e). A thumbnail is a miniaturization of an image. Here they are displayed in a group to enable previewing of multiple images. The images can be enlarged for ease in viewing by simply clicking with the cursor directly on the thumbnail (Fig. 3-f). The discussions can be found by clicking on the hypertext links to any discussion, history, pathological findings or diagnoses which are available with that case.

Another aspect of this web site that could be potentially very useful is the link to questions and evaluation forms for physicians interested in obtaining continuing medical education (CME) credits. The courses are worth only one unit, but they are easy to access at any time. There is no fee for military and federal government physicians, and only a small fee for civilian pathologists.

3.5 Commercial Databases

Chapman & Hall, a publishing company based in New York City now offers Pathology Atlases containing a collection of approximately 55,000 images available on CD-ROM. This exists as the largest source of digitized pathology slides available. Chapman & Hall advertises 21 image libraries on CD-ROMs, each representing a specific organ system. Eighteen image libraries are currently available for order today, including blood, lung cytology, skin/pigmented lesions, pituitary, and thyroid pathology, with three more modules (ovary, prostate, and benign lung pathology) in the works. It should be noted that neither the nervous system nor the muscular system are represented in the collection. Each atlas contains anywhere from 700 to over 16,000 images with supporting descriptive legends. In addition, lists of references and any applicable textual/lab data, including clinicopathological findings, immunology, cytogenetics, and chemistry are available. Lecture notes geared to residents and students are also included.

The CD-ROM opens with a main menu offering access to the information in six different ways: by disease atlas, feature atlas, references, feature definition, disease definition, or lecture (Fig. 4-b). The user can choose the disease atlas to view images organized by disease (Fig. 4-c). Clicking on a disease category of interest, the first color image appears on the screen with a brief description at the bottom (Fig. 4-d). Clicking on the image will bring up a zoomed, full-screen image that is more than twice the original size (Fig 4-e). In order to view many images quickly, there is a button titled "thumbnails" which brings you to a screen with eight small images and their associated text (Fig 4-f). There are also direct links to a list of related references, an extended definition of the disease (Fig 4-g), and other pictures and cases.

With the feature atlas, images can be accessed by first electing certain descriptive terms in the main menu (Fig. 4-h) and then choosing from more specific submenus that appear over this main page (Fig. 4-i). This enables you to compare an unknown specimen to slides defined in the module. The objective is to make comparisons quick and easy.

Another useful tool available in the pathology atlases are the lecture series. After choosing a subject from the lecture menu, information is displayed at the resident or medical student level (Fig. 4-k) with links to related images (Fig. 4-l).

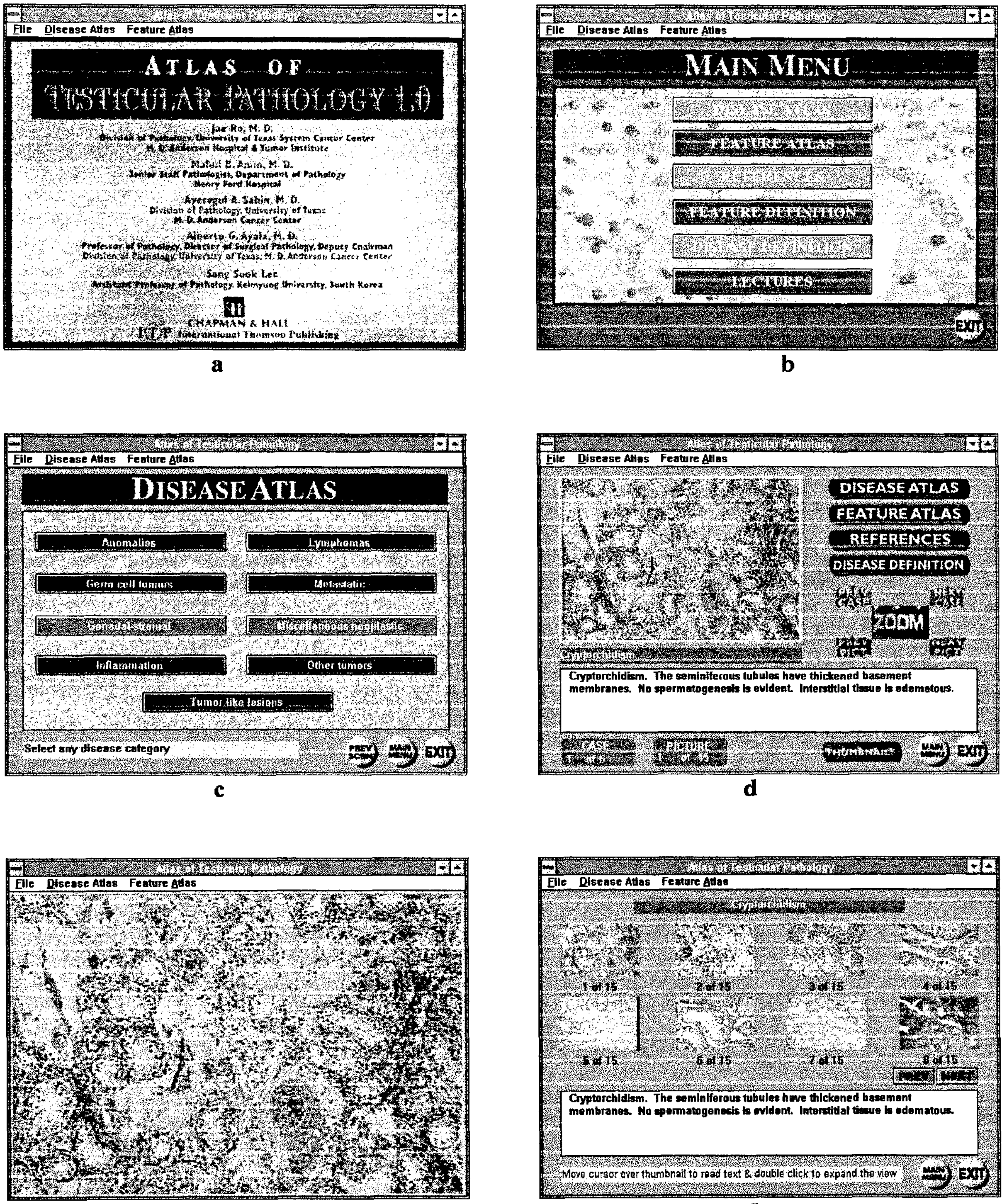


Fig. 4 - Major Frames from Pathology Atlas, Chapman & Hall.

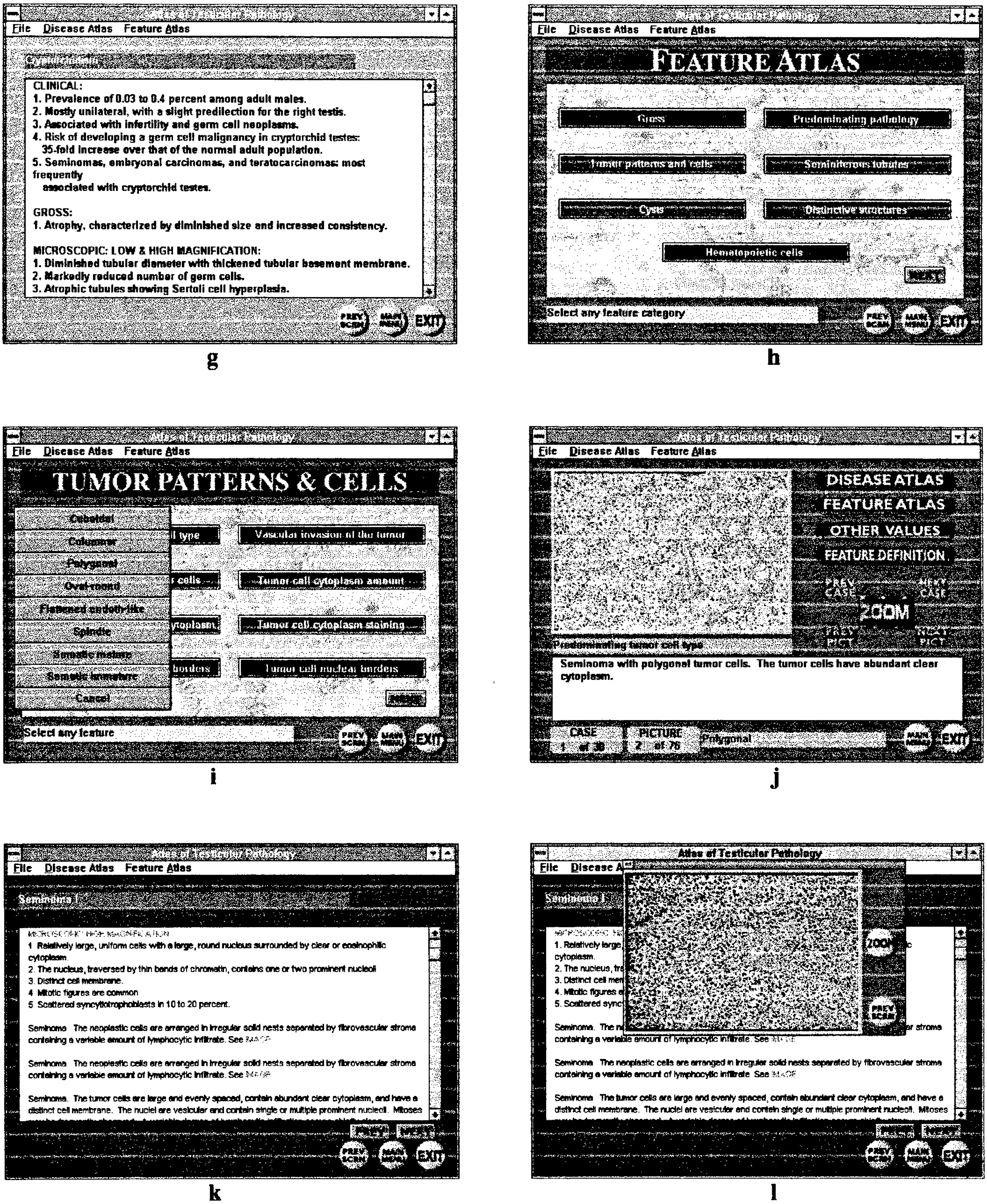


Fig. 4 - Major Frames from Pathology Atlas, Chapman & Hall.

Mosby Multimedia, a publication company based in England, has produced a single CD-ROM which is modeled on the textbook *Pathology,* by Alan Stevens and James Lowe. This CD-ROM serves as an educational tool to help students learn more about basic pathological processes and clinical pathology. The user can turn the pages of this virtual book by clicking on "curled corners" of the page. Each page is accompanied by text as well as diagrams, charts, illustrations or photomicrographs. The zoom feature allows the user to view an image magnified (Fig. 5-c). It works very much the same as an ordinary microscope: only a portion of the enlarged image can be seen at once and as you move the cursor up with the mouse you scan down the image. This orientation is inverse to that customary to that used in the U.S.A. It also applies to moving around the image from left to right, and right to left. This makes viewing an image at higher magnification very intuitive to those who are already accustomed to the standard light microscope.

Many colorful 2-D and 3-D animations are included throughout the "book." In certain cases sound files are added, such as with the three introductory movies that explain how to use the features of the program. *Pathology* is also set up to "speak" out the main text and figure descriptions throughout the CD-ROM, as long as you have the right extensions installed on your computer.

The search features allow you to quickly browse through chapters of interest, as well as locate more specific material from the extensive index. Figure 6-d shows one browsing tool called the "flickbook." The chapters are displayed by number, and the titles appear at the bottom as you move the cursor across each button. Clicking a button takes you directly to that chapter. There is a slider beneath the flickbook's main display window which you can move from side to side to quickly preview pages of that chapter. When you find a page of interest, releasing the mouse takes you directly to that page. You can place a bookmark on that page, allowing you to return to that page from anywhere in the book.

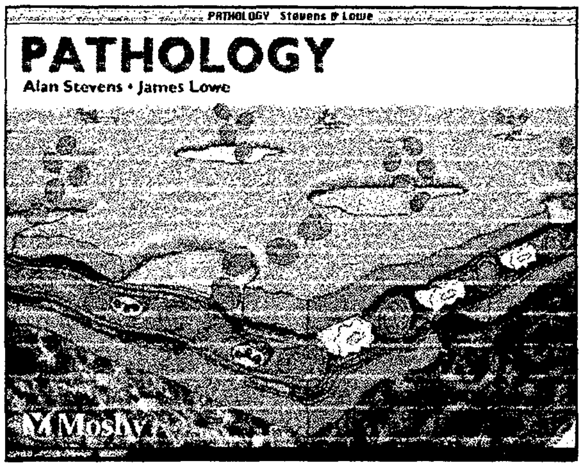
In order to locate pages by subject, you can use the search feature, shown in figure 5-e. There is an alphabetical listing from which you can search, but you may also type a word or phrase into the box at the bottom of the display to view any related material. A small defect was detected in this search feature. Clicking on some items in the alphabetical listing links you to the wrong page. Mosby Multimedia has been informed of the defect, and are working to remove the bug. The newest version 1.1 was released after the same problem was encountered in version 1.0, but apparently the problem was not completely solved.

3.6 Kensal Database

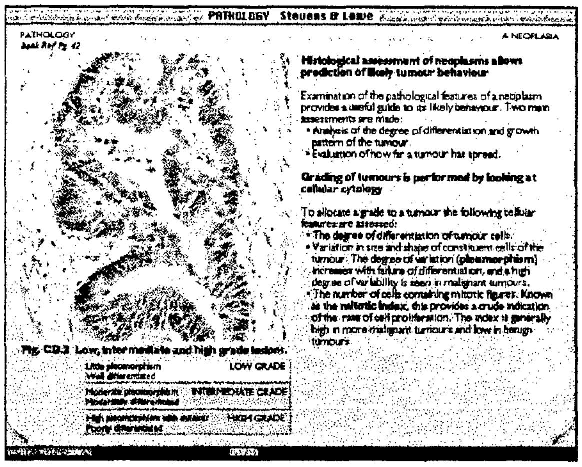
The Kensal Corporation is currently developing an interactive CD-ROM "virtual microscope" to be used by medical school students and pathologists. The CD-ROM uses full-coverslip lensless scans and the accompanying high-magnification images to give the user the impression that he/she is observing a glass slide using Kensal's telepathology workstation. Approximately 264 images comprise the CD-ROM, of which 24 are full-coverslip guide images.

The user has two options with the virtual microscope. First, the user can explore the full-coverslip image at full resolution by moving about the coverslip. Second, he/she can select one of the regions of interest for higher magnification and explore it in a similar manner.

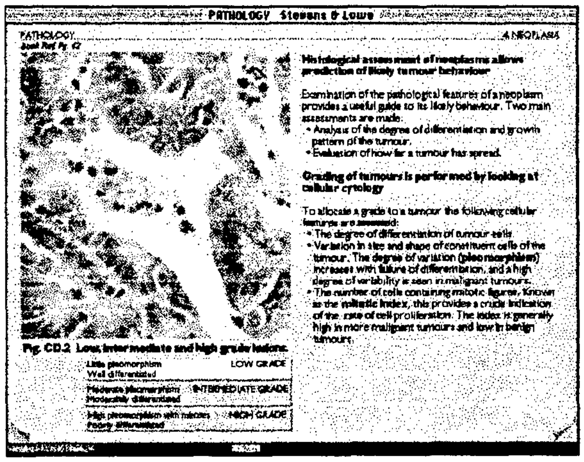
There are three ways the virtual microscope user can access images. The first method is by using the Model Search, in which a human model appears on the screen with several different organ systems to choose from (Fig. 6-b). The user has the option to change from a male to a female, and to rotate the model from a front view to a rear view. Clicking on an organ system will bring up one guide image pertaining to that system (Fig. 6-c). From that screen the viewer has the option to obtain the diagnosis/information about the tissue (in the form of text and voice annotations), or to view a full-screen image of the tissue (Fig. 6-d). In addition, located on the



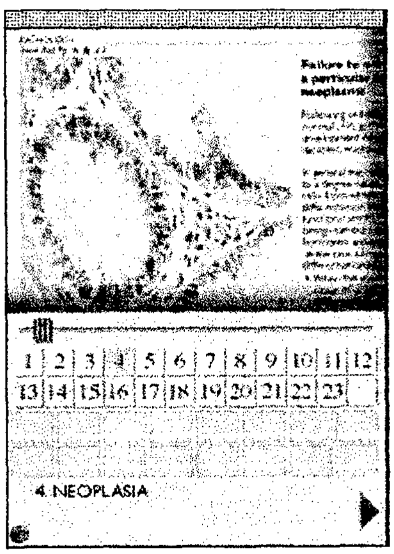
a



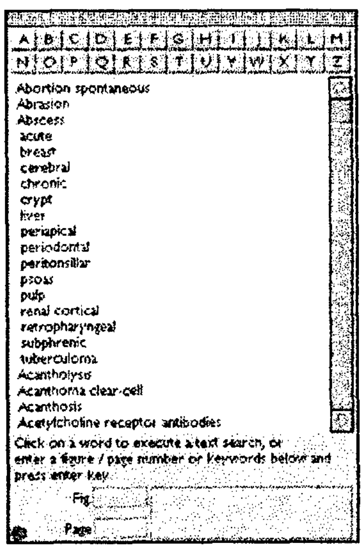
b



c



d



e

Fig. 5 - Major frames from Pathology CD-ROM.

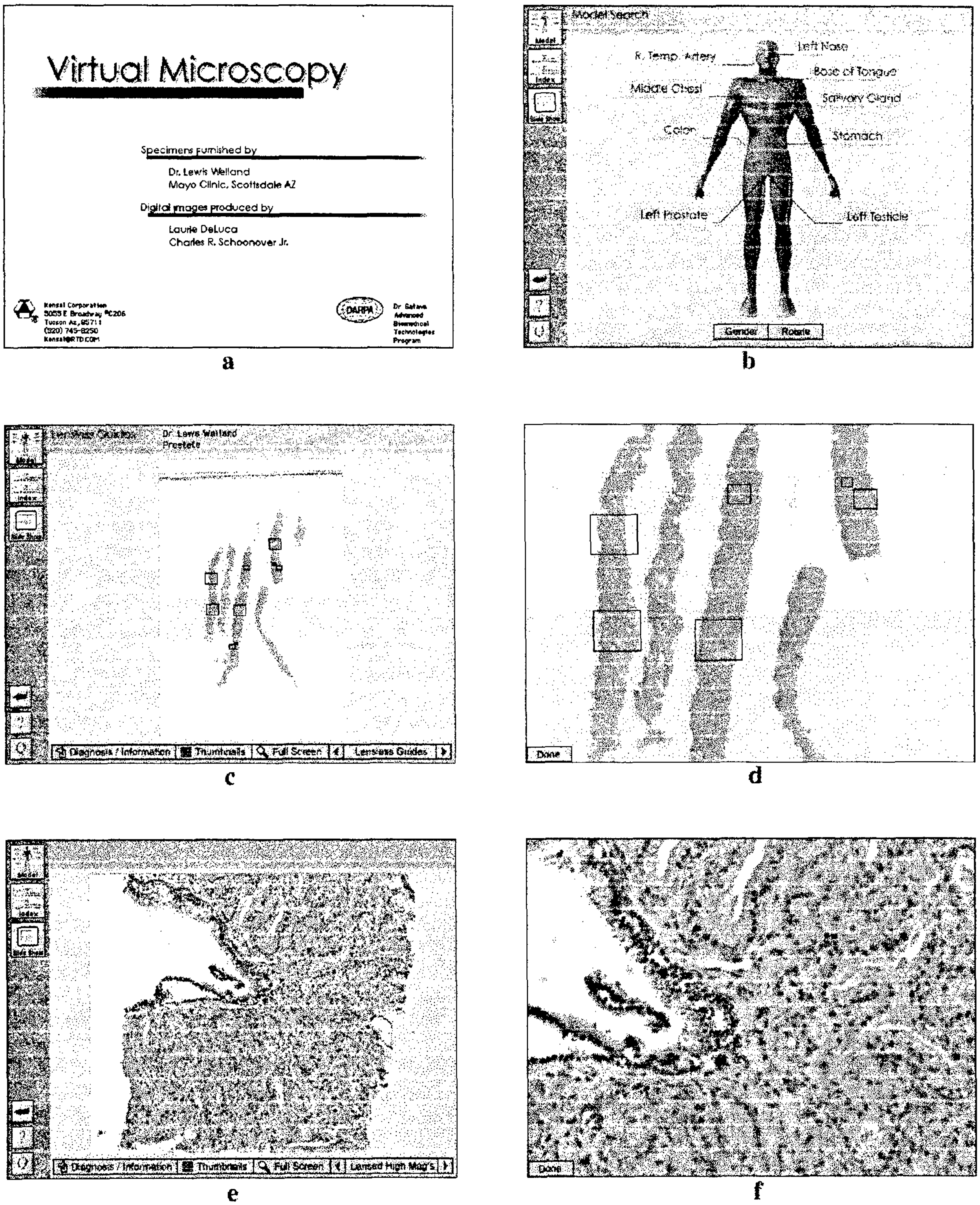


Fig. 6 - Major frames from Virtual Microscope CD-ROM produced for U.S. Army Medical Research Command.

guide image are markers indicating the regions of interest (ROI). The size of the markers indicate the degree of magnification. Larger markers are affiliated with low power magnifications, while small markers indicate higher magnifications of only a small area of tissue. By clicking on one of these markers, a high magnification image will appear for that location (Fig. 6-e). Again, the diagnosis/information may be obtained for that image, as well as bringing the image to the full scale of the screen (Fig. 6-f).

The second method of accessing images is through an index. In the index, all of the images are listed alphabetically by topology and by organ system, offering two ways to access the image of your choice. Clicking on the desired image will bring you to that specified guide image. The same process for accessing the high magnifications is followed as is done with the Model Search.

Lastly, the user can choose the "slide show" section, in which they may choose images from an index to be saved and played back automatically or saved to a disk. This section includes the high magnification images and any voice annotations that go along with the images.

3.7 Summary

Each image database reviewed differs in the total number of images available, as well as in how the images are distributed among organ systems. Kensal Corporation has taken a closer look at the microscopic image counts for each database to distinguish between which institutions are focused in certain areas and which are lacking in other systems all together. A series of pie charts representing the microscopic images has been developed for each organ system (Fig. 7). This has been done so that the viewer may see how each image database compares with the other's representation of the systems of the human body.

After counting the number of microscopic images in each organ system category, percentages from each database were calculated and entered into the graphs. Therefore the largest slice of the pie represents the database which offers the greatest emphasis of microscopic images in that system when compared to the other databases. The viewer must take into consideration that these pie charts are only showing the "relative emphasis", and not the total number of microscopic images. For instance, Chapman & Hall has the greatest number of microscopic images in the female reproductive system, estimated at 1,326 images. The pie chart for this system, however, shows the AFIP with the largest section of the pie for that organ system. While the AFIP only has 126 images in the female reproductive system, this is a large percent of their 347 images (37%). Chapman & Hall's 1,326 images of that organ system represent only 3% of their total 55,000 images. Therefore, Figure 7 is a tool that can be used to figure out which databases emphasize certain systems. The viewer should remember that this chart in no way compares the total number of images available (this information can be found in the chart on pages 11-14).

Chapman & Hall clearly contains more images overall than any other database. Due to the high volume of slides though, it was difficult to determine the exact number of microscopic images available in the collection. Sanjiv Patel, an engineer and programmer for this collection, stated that no exact count of microscopic images was available from the publisher at this time. He did estimate, however, that between 80 and 90% of the slides are microscopic images, with the remaining images being gross specimens. No diagrams or x-rays are included in the collection. To verify this estimate we did a count in two of the modules, The Atlas of Testicular Pathology and The Atlas of Thyroid Pathology. We found that these two atlases contained 77% and 99% microscopic images, respectively.

WebPath contains the second largest collection of microscopic images, with just under one thousand. Out of these images, each organ system is represented reasonably well. The areas that

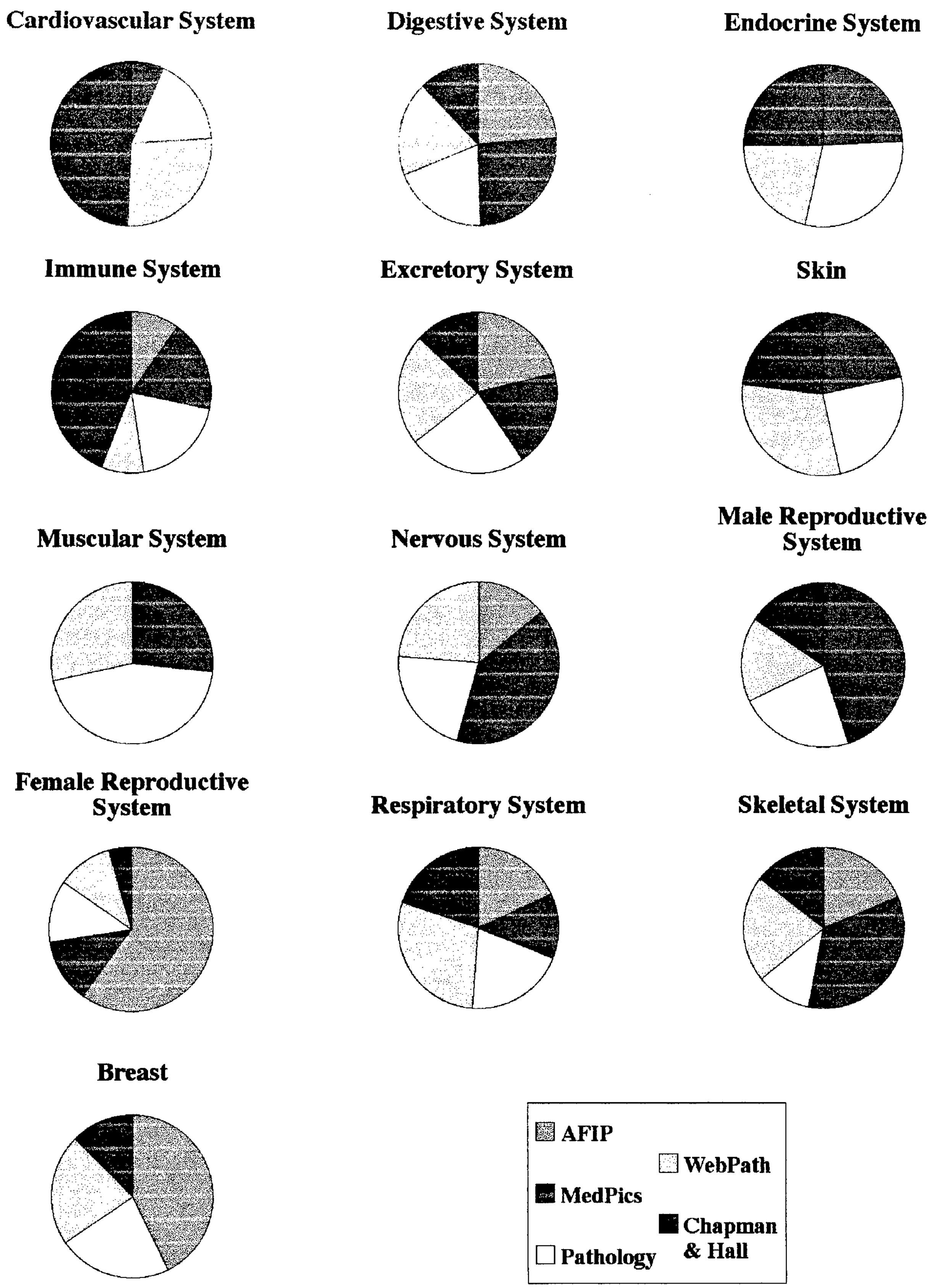


Fig. 7 - Relative Emphasis of Microscopic Images per Organ System.

TABLE 2 - Image Libraries

Features Surveyed			
Publisher	University of California, San Diego (LaJolla, CA)	University of Utah (Salt Lake City)	Armed Forces Institute of Pathology (Washington D.C.)
	Department of Pathology	Department of Pathology	Center for Advanced Pathology
Internet Address		http://www-medlib.med.utah.edu/WebPath/webpath.html	http://www.afip.mil
Name of Product	MedPics	WebPath	Armed Forces Institute of Pathology Web Site
Number of Disks	1 disk	1 disk	NA
Price	$79	$50.00	Free
Available	Yes	Yes	On the Internet only
Demo Disk	No	No	No
Operating System	Windows 3.1 or Macintosh System 7.0	Macintosh or PC compatible	Macintosh or PC Compatible
Minimum System Requirements			
RAM	5MB		
Processor	386SX 33MHz		
Hardware Requirements	Double-speed CD-ROM drive, mouse	Mouse or compatible positioning device	Mouse
Monitor Requirements	8-bit color (640 x 480)	800 X 600 resolution	
Software Requirements	None	Web browser	Web browser
Total Images	580	2007	347
No. Gross Images	172	895	2
No. Microscopic Images	350	997	338
No. Diagrams and Illustrations	31	56	0
No. Electron Micrographs	24	57	0
No. X-Rays	3	2	7
No. Images per Organ System			
Cardiovascular System	11	125	0
Digestive System	72	150	63
Endocrine System	21	53	0
Immune System	43	58	23
Excretory System	34	112	35
Skin	16	66	0
Muscular System	9	28	0
Nervous System	47	78	16
Reproductive System, male	17	18	0
Reproductive System, female	27	66	126
Respiratory System	23	146	31
Skeletal System	30	54	16
Breast	0	43	28
Textual Data	Supporting text fields on specimen, preparation, view, key features, and diagnosis. Provides a list of learning objectives, and a directory of images for each tissue/organ system.	Supporting text fields on specimens.	Images have hypertext links to case histories and diagnoses.
Color	Yes	Yes	Yes
Voice	No	No	No
Zooming Capabilities	No	No	No
Technical Support	No	No	No
Indexing Method	Images categorized by organ system.	Images categorized by general pathology, organ system pathology, laboratory exercises, examinations, mini-tutorials, clinical pathology and histopathology.	Images categorized by departments of the Center for Advanced Pathology.
Coding/Standardization	No	No	No
Years on the Market	5	3	
Future Developments	Version 2.0 coming fall, 1996	Regular updates to Web site	Regular updates to Web site
Target Audience	Medical students	Medical students	General public, pathologists
Digitization (direct/35mm slide)		Images scanned from kodachromes.	
Additional Features which Distinguish it from the Competition	Program offers two modes: the standard teaching mode, or a self-testing mode to let users identify randomly presented images without on-screen descriptive cues. List of features which hide/display a graphic overlay on the image identifying the structure.	Description with illustrations of diagnostic techniques and procedures. Multiple choice examinations and sample essay questions for self-testing. Available on CD-ROM and via the World Wide Web. Web site links to other medical education resources.	Some images are presented as courses, which can be viewed to obtain Continued Medical Education units of credit. Offers additional listings of CME courses and educational programs, and includes on-line registration forms.

TABLE 2 - Image Libraries (continued)

Features Surveyed			
Publisher	Chapman and Hall Publishing (New York City)	Mosby Multimedia (Eden Prairie, MN)	Kensal Corporation (Tucson, AZ)
Internet Address	http://www.chaphall.com/chaphall.html		http://www.amug.org/~kensal
Name of Product	Pathology Atlases on CD-ROM (1.0)	Pathology	Virtual Microscopy
Number of Disks	18 disks		1 disk
Price	$2,750	$59.95	
Available	Yes	Yes	Coming soon...
Demo Disk	Yes	30 day preview	Yes
Operating System	Windows 3.1 or higher	Windows 3.1 or Macintosh System 7.0	Macintosh
Minimum System Requirements			
RAM	8MB	8MB	8MB
Processor	486SX processor	68030 Processor or better	
Hardware Requirements	50 MHz double-speed CD-ROM drive, mouse	Double speed CD-ROM drive	
Monitor Requirements	16-bit color (640x480)	16-bit color (640x480)	8-bit color (640 x 480)
Software Requirements	None	None	None
Total Images	55,038	799	about 300
No. Gross Images	8253	343	
No. Microscopic Images	46785	316	
No. Diagrams and Illustrations	0	122	
No. Electron Micrographs	0	7	
No. X-Rays	0	11	
No. Images per Organ System			
Cardiovascular System	10593	25	
Digestive System	4556	48	
Endocrine System	2890	23	
Immune System	13813	41	
Excretory System	3110	37	
Skin	2342	17	
Muscular System	0	14	
Nervous System	0	23	
Reproductive System, male	778	8	
Reproductive System, female	1326	24	
Respiratory System	4689	33	
Skeletal System	1696	9	
Breast	1101	14	
Textual Data	Supporting text fields on specimens, extended descriptions of clinical and pathological findings and diagnoses, and lists of references.	An interactive text book, including both a brief and an expanded version of the text. Underlined sections of text are linked to cross-references.	
Color	Yes	Yes	Yes
Voice	No	Yes	Yes
Zooming Capabilities	Full screen option	Yes	Full screen option
Technical Support	Hot line number, instruction booklet	Mosby Multimedia Tecnical Support Hotline	Instuction booklet
Indexing Method	Images categorized by specific disease, or feature.	The CD-ROM is organized like the original text book, indexed by chapter title as well as by feature. The search screen allows you to select items by word, phrase, or page number.	Images listed by organ system.
Coding/Standardization	No	No	SNOMED coding
Years on the Market			NA
Future Developments	Three new modules expected in the next year.		
Target Audience	Pathologists	Medical Students	
Digitization (direct/35mm slide)			Direct digitization/scanning
Additional Features which Distinguish it from the Competition	Lecture notes with hypertext links to images are avilable.	Includes introductory videos that explain how to use the program. Colorful 2-D and 3-D animations. Images can be magnified further for viewing. Most photographs, diagrams and animations are accompanied by buttons to display labels and areas of interest.	"Virtual microscope" allows you to first view an entire slide, from which you can choose regions of interest for higher magnification.

contain the most images are the respiratory system and the cardiovascular system, while the areas with the least number of images are the male reproductive system and the muscular system.

Mosby Multimedia's *Pathology* CD-ROM also represents each of the thirteen organ systems. The largest percentage of their images lies within the digestive system, and the least within the male reproductive system. Likewise, MedPics does a good job of representing all but the breast in their CD-ROM.

The AFIP, on the other hand, leaves five systems unrepresented with no images: the cardiovascular system, the endocrine system, skin, the muscular system, and the male reproductive system. The female reproductive system alone is represented by over a third of their images.

3.8 Acknowledgments

The previous table lists detailed data of these six databases. The information was collected by Kensal Corporation through contacting vendors and creators, then reviewing the contents of each program. We do advise that readers interested in purchasing any of the CD-ROM's verify the price quotes, availability, and current status of the product.

We would like to acknowledge the creators, engineers, and publishers of the databases for their help in this review. Financial support is being provided by DARPA via the U.S. Army Medical Research and Materiel Command.

4. HOSPITAL INFORMATION SYSTEMS

Information systems (IS) are defined as a set of people, procedures, and resources that collect, transform, and disseminate information in an organization. The goal of IS is the production of accurate and timely information - products for end users. An IS should also produce feedback about its input, processing, output, and storage activities to determine if established performance standards are being met.

The Hospital Information System (HIS):

- ♦ manages hospital finances and resources
- ♦ furnishes decision support through ad-hoc reporting
- ♦ automates office work at all levels
- ♦ tracks/manages patient data, care and billing
- ♦ establishes inter-communication and data-exchange between: other hospitals, insurance and billing agencies, clinics, laboratories, nursing stations, other information systems and data repositories, wired or wireless instrumentation and printers, technologists, and physicians in the hospital or elsewhere.

The primary motivation behind HIS implementation is cost savings. An inherent secondary benefit is improved health care and lower error rates due to improved organization and communication.

4.1 HIS Modularization

Since more feature-sets and end-users have been added to the HIS domain over the years, the HIS has become increasingly modular with outgrowths in both clinical and laboratory information systems (CIS/LIS). Savings through better organization, increased automation, and faster order/result turnaround are the compelling reasons for outgrowth and modularization. Each sub-IS module not only acts as a store-and-forward/fault-tolerant repository of data within the central HIS. This eliminates communication traffic problems between nodes. Data being transmitted is first stored and waits until bandwidth is available for transmission. This does two things, 1) frees up the end-user's terminal to work on other things while the data is queued and, 2) ensures fault-tolerant delivery of data in case of temporary disconnection during transmission--data can be resent. Each module also provides domain-centric ad-hoc feedback to inquiring administrators who must report to hospital enterprise leaders and government regulating agencies. The six related and overlapping systems within the health care field are:

- Management Information Systems
- Financial Information Systems
- Telemedicine Information Systems
- Knowledge Systems
- Public Health Information Systems
- Research Systems

4.2 HIS Security

Throughout the chain of hospital rank and command, layered need-to-know based security protects sensitive information within the HIS. While nursing stations can prepare work lists based on patient needs, they cannot view executive level reports. While the admissions-desk can see bed availability in real time, they cannot change laboratory results. While physicians can submit pharmaceutical and laboratory orders, they cannot admit, discharge, or transfer (ADT) patients. Security is necessary to protect patient privacy, control efficiency and order, as well as, prevent accidents, fraud, or deliberate reprisals and sabotage against the hospital enterprise.

4.3 A Powerful Management Tool for Strategic Planning

Since hospitals are in a market driven by cost containment, enterprise leaders are constantly looking for ways to minimize the cost of care in all service areas. The HIS, when well implemented, can be used as a powerful management tool to guide decision- making. The ideal HIS provides rich insight and decision support for the optimal financial structuring of the hospital.

"Managed care" was cited as the most significant force driving increased computerization in health care.

Managed care continues to represent a cost-centered approach to computing, as 49 % of more than 1200 surveyed in the 1996 HIMSS/HP Leadership Survey said accessing and analyzing financial information for better management of overall costs is the most important advantage of computer technology for managed care.

However, clinical concerns [of managed care providers] are becoming more prominent, as 45% of those surveyed said accessing clinical information (data and images) from specialty services was most important. (For more information, see Appendix A, 1996 HIMSS/HP Leadership Survey.)

4.4 Cost-Containment Pressures Drive HIS Innovation and Integration

In the near future, HIS will likely be linked to a Community Health Information Network (CHIN) to help minimize costs within a group of collaborating health-care providers. In addition, research by the National Information Infrastructure-Health Information Network (NII-HIN) Consortium is underway to develop standards to provide transparent linkages between CHINs through a national information infrastructure. Finally, there is some movement towards a Global Health Network (GHNet) which is focused on the critical role of prevention in reducing health care costs through rapid, accurate transmission of information. Other innovations include computer-based order/results entry and point-of-care reporting.

4.4.1 Community Health Information Networks

"CHINs are community-wide electronic networks of health care providers, medical facilities, payers, pharmacies, and other health care support companies that allow the sharing of patient medical and financial data in a more efficient manner. CHINs can also support the sharing of radiological images and live telemedicine. A regional CHIN promises to improve the quality of patient care and lower the cost of health care in the community." Before a hospital can be integrated into a CHIN however, it must support a Computer-based Patient Record (CPR) that can be transparently passed to other hospital HISs of likely dissimilar implementation. Many hospitals still keep much of the patient record on paper in a folder labeled with the patient's unique HIS index number. Recently, enormous industry and media attention have been focused on the CPR. Despite this, hospitals in general are hesitating to implement a CPR. CHINs are thus just emerging in the health care industry but will play a significant role in the near future of health care.

4.4.2 Computer-based Patient Record (CPR)

The CPR concept is fundamentally a computer-stored collection of health information about one person linked by a personal identifier. The CPR or the "electronic patient record" are terms used by vendors interchangeably but refer to different levels of computerization. Clarification regarding these levels has been outlined by the Medical Records Institute (MRI), founded on the principle that the future of health information technology lies in the successful creation and implementation of electronic health record systems. Although in fact five levels have been defined, only the first two levels have been achieved--levels 3 through 5 are not felt to be possible for some time. The five distinct levels of computerization for patient information systems has been outlined by MRI as follows:

Level 1: Automated Medical Records

Are paper-based medical records with as much as 50% of the printed content computer generated. Level 1 automation within the hospital environment is focused around the following functions:

- ADT (Admission/Discharge/Transfer) systems
- Improved capture of patient information through digital dictation systems
- Patient accounting and its linkage to clinical information

- ♦ Departmental systems (i.e., Radiology Information Systems, Laboratory
- ♦ Information Systems, Pharmacy Information Systems, etc.)
- ♦ Order Entry/Results reporting (discussed in section 1.4.3, below)

Other innovations parallel to the paper-based medical record are nursing/bedside computing (discussed in section 1.4.4), implementation of an enterprise-wide master patient index, the linkage of various parts into an enterprise-wide network, the development of interface engines and imaging.

Level 2: Computerized Medical Record System

Level 1 automation does not solve the space shortage in record storage, nor create an electronically available record. A level 2 computerized-medical record system (or document imaging system) allows paper-based medical records to be created, then scanned, and indexed within a computer system with the same automation functions as level 1. Optical Character Recognition (OCR) or Intelligent Character Recognition (ICR) do not fit into level 2 automation since the scanned documents are stored on optical disks as unchangeable images, not ASCII-based data-sets. Level 2 is the only method in existence as of this writing to computerize the medical record in a paperless system.

Level 3: The Electronic Medical Record

The level 2 computerized medical record has basically the same structure as the level 1 paper-based medical record. The level 3 electronic medical record has the same scope of information in level 2 but the information is rearranged for computer use. While the level 1 paper-based records system is a passive storage device, level 3 can provide interactive aiding of the decision-making process by knowledge coupling, providing decision support, and many other functions. Level 3 requires a secure enterprise-wide infrastructure for appropriate capture, process and storage of patient information.

Level 4: Electronic Patient Record Systems (also called Computer-based Patient Record Systems)

The patient record has a wider scope of information than the medical record. It combines several enterprise-based electronic medical records concerning one patient and assembles a record that goes beyond the enterprise-based retention period.

Level 5: The Electronic Health Record

The more comprehensive collection of an individual's health information is the level 5 electronic health record. It differs from the electronic patient record in the unlimited amount of health information captured by caregivers regarding a person. It includes wellness information possibly captured by the individual or parents, therapists, etc., including data for example on behavioral activities such as smoking, exercising, dietary and drinking habits. The electronic health record is maintained through cooperation between the individual who controls his or her health information, and the caregiver.

4.4.3 Computer-based Order and Results Entry

Savings from an order entry module in the HIS include form costs, lost charges, a significant reduction in "telephone tag" between nursing and ancillary departments, elimination of mistakes due to legibility, and the establishment of controls for accountability within the hospital. A results reporting module allows entry of both brief and verbose "status" reports on received orders.

4.4.4 Computer-based Point of Care

A recent HIS appendage and innovation for cost-savings are computerized point-of-care systems (CPSI). HIS vendor CPSI markets a "Chart Cart" --a portable PC on a medicine-cabinet cart with a touch sensitive screen and bar code reader, all wirelessly connected to the HIS--which allows Nursing Services personnel to enter information into the HIS at the patient's bedside. Clerical functions are automated and duplicate entry of information into nursing documents is eliminated. Charges for administered medication can be billed immediately by using the keyboard and bar code reader to scan the medication container.

HIS vendor MEDITECH also has a 14-ounce, hand-held personal digital assistant (PDA) for computer-based point of care. End users of this device are nurses, nurses aides, and therapists. The PDA holds data for 10-20 patients and keeps track of the "whereabouts of physicians". When a nurse's shift begins, the nurse downloads patient records into the PDA and then administers to the need of the patients. During the shift, the nurse can operate the PDA with one hand's thumb--to see orders and record results--while administering care with the other hand. After the shift, the data in the PDA is uploaded into the HIS.

4.5 Health Care Information System Priorities

The most important IS priorities for health care organizations are upgrading their IT (Information Technology) infrastructure and integrating systems in a multivendor environment. Reengineering to a patient-centered computing environment is also receiving priority attention from health care organizations. And organizations are following through by completing these projects.

4.6 Computer Imaging In HIS

Computer imaging is relatively new to the HIS. Several outgrowths are currently integrating images and text in stand-alone modules (i.e., radiology, paperless-office, and telemedicine modules). However, there is no standard way to integrate the text-based medical record and related digital image-based entities together for call-up throughout the HIS, much less across hospitals or CHINs. Thus, a tremendous amount of work yet lies ahead to create, what might be coined as, the "Graphical Patient Record" (GPR).

While standards do not exist yet meshing text and large binary objects like images for HIS-wide access, Los Alamos National Laboratories (LANL) has recently announced TeleMed which contains an experimental GPR focused currently in teleradiology. TeleMed, based on a distributed national radiographic and patient record repository which could be located anywhere in the country, is designed to assist doctors in treatment planning through viewing patient treatment histories and associated radiographic data. These data can be viewed simultaneously by users at two or more distant locations for consultation with specialists in different fields. LANL claims that TeleMed "is the first to provide transparent access to patient record components over a Wide Area Network (WAN), building the complete patient record from various partial records and displaying that in an integrated manner to the healthcare provider."

Industry standards are needed for seamless integration of images throughout the HIS. Once again, no standard exists which integrates text and images across the entire HIS as of this writing; however there are several SDO's (Standards Developing Organizations)--who have good

foundations and the technical resources--developing such a standard. These groups and their progress will be described below under section 2.2, Medical Informatics Standards Groups.

4.7 Reports Available on HIS

Available HIS reports are endless and their titles vary from vendor to vendor. Often, vendors will tailor report content and structure to the needs of the hospital. Thus, unlike the IRS or other government branch, there is little report standardization, except in the insurance billing modules and in the reports destined for accreditation overseers.

Often provided in the various HIS modules, is the ability to generate ad-hoc reports; thus, in addition to the "canned" reports unique to a particular HIS vendor, reports of any content or structure can be generated through Standard Query Language (SQL) inquiries on a database. However, the HIS end-user must learn how to use the SQL interface and the semantics of the query language before useful reports can be generated.

As mentioned in section 1.4.2 earlier, an extreme interest in moving to a paperless reporting mechanism has been manifest in many hospital enterprises, due to cost savings. Most of the HIS vendors are just now beginning to offer the "Level 2" document imaging ability. "Level 3" is highly desired, but requires physical and logical integration across disparate facilities and computer systems, with nearly a unique solution for each integration case. To understand the barriers to enterprise-wide electronic report exchange, the physical and logical architecture of the HIS will be discussed in the next section.

4.8 HIS Related Topics

Below the HIS application's layer is a complex data storage and exchange network. These networks are based on numerous standards necessary to bring order to the physical mediums used for communication, the inter-node communication protocols, the physical/logical interface to the computing platforms, the applications level communication management, and the monitoring and reporting instrumentation. The interconnected machines themselves make up a heterogeneous and distributed computing environment. Careful understanding of the standards at all levels are thus needed before attempting to add bilateral information exchange nodes to an existing HIS, lest the delicate system-balance of the HIS be upset.

There are many standards groups who's specifications are being used to implement the HIS. At the messaging level--the level where HIS nodes exchange information related to the health-care industry--various standards groups, many driven by HIS vendor innovation, have been working together to build the expanding field of medical informatics. At the lower hardware level, Institute of Electrical and Electronic Engineers (IEEE), International Standards Organization (ISO), International Telecommunications Union - Telecommunications (ITU-T), American National Standards Institute (ANSI), et al, have published networking specifications in circulation for years, used in HIS implementation. Newer negotiated-multiband technologies such as Asynchronous Transfer Mode (ATM) for information interoperability are also being used in some HIS implementations.

4.8.1 Medical Informatics

Biomedical Informatics is an emerging discipline that has been defined as the study, invention, and implementation of structures and algorithms to improve communication, understanding and management of medical information. The end objective of biomedical informatics is the coalescing of data, knowledge, and the tools necessary to apply that data and knowledge in the decision-making process, at the time and place that a decision needs to be made.

The focus on the structures and algorithms necessary to manipulate the information separates Biomedical Informatics from other medical disciplines where information content is the focus.

4.8.1.1 sci.med.informatics Newsgroup

The medical informatics USENET newsgroup (accessed at the above address) is open to the Internet public. As stated in their Charter: The focus of this newsgroup will be the discussion of the grand challenges facing medical informatics today (and tomorrow). Appropriate topics include, but are not limited to:

* Medical Information Standards
* Medical Informatics Training
* IAIMS (Integrated Academic Information Management Systems)
* **Computerized Medical Records**
* **Clinical Information Systems**
  (including radiology, laboratory, pharmacy, nursing, etc.)
* **Physician Order Entry Systems**
* Computer-Aided Instruction
* Medical Expert Systems
* Nursing Informatics
* Announcements of Interest, e.g. **conferences, journals, societies**
* National Library of Medicine
* Health Information Networks
* **Medical Software Reviews**
* Research Funding Opportunities
* Policy Making
  (including procurement and certification of medical software)
* **Medical Software Engineering**
* Cultural/Sociologic Changes
* **Medical Software Security**
* **Telemedicine**
* Veterinary Informatics

4.9 Medical Informatics Standards Group

The term "standards" includes standards developed by accredited standards organizations and other categories of organizations who are affecting, or working on, technical, procedural, and systems standards, guidelines, professional protocols, minimum requirements, as well as industry practices necessary to enable the computer-based record system of the future to function. From this perspective, there are seven categories of organizations involved in the process:

1. Major standards organizations who develop application standards for health care
2. Professional societies involved in standards creation
3. Trade associations
4. Government organizations
5. Industry consortia
6. National players
7. Standards organizations for base standards

The Medical Records Institute provides an *International Directory of Organizations: Standards and Developments in the Creation of Electronic Health Records* which lists over 160 different groups working on standards in health care throughout the world; outlining their current projects, publications and reports.

One of the largest components in the HIS standards work in progress is the design effort taking place to specify how digital messages should be exchanged between HIS computer systems and what they should contain. These messages encapsulate information ranging from ADT updates to lab-results data. The messaging structures implemented in HIS systems today are analogous to the different foreign languages and/or dialects spoken in various regions of the earth--from the global HIS market perspective, every vendor has its own unique standard or, more frequently, *interpretation* of a local recognized standard (i.e., HL7, discussed later). Since a substantial technical investment is required to enable one vendor--faced with appending modules on to HIS systems from other vendors--to speak all these languages and dialects, convergence to a common language--or messaging standard--is the drive behind the messaging Standards Developing Organizations (SDOs) today.

4.9.1 The Message Standards Developers Subcommittee (MSDS)

In 1991 there were at least six organizations developing health care messaging standards, of which three were accredited by the ANSI. During that year, the European standards agencies asked ANSI to clarify with whom they could coordinate health informatics standards. As a result, ANSI formed the Health Informatics Standards Planning Panel (HISPP) to coordinate the development of health informatics standards. HISPP's membership includes system vendors, professional organizations, SDOs, and various users of standards.

In turn, HISPP formed a subcommittee of its members who were standards developing organizations. This is the Message Standards Developers Subcommittee (MSDS). The members of MSDS are SDOs developing health care message interchange standards. The objective of the MSDS is to achieve harmonization of the standards that SDOs develop.

4.9.1.1 MSDS Member Organizations

ASTM:	American Society for Testing and Materials
DICOM:	Working groups of American College of Radiology (ACR) and National Electrical Manufacturers Association (NEMA)
HL7:	Health Level Seven
IEEE:	Institute of Electrical and Electronics Engineers Medical Data Interchange Working Group
NCPDP:	National Council of Prescription Drug Pharmacies
X12N:	Insurance Subcommittee of ASC X12

The MSDS formed the Joint Working Group for a Common Data Model (**JWG-CDM**) as an open standards effort to support the development of a common data model that can be shared by developers of health care informatics standards. The IEEE Committee has secretariat responsibility for the activities of the JWG-CDM. Thus, for all practical purposes, the IEEE Medical Data Interchange Working Group and the Joint Working Group for a Common Data Model are identical. The acronym JWG-CDM refers to these groups.

On June 6, 1994 the IEEE Standards Department made available the initial draft of the JWG-CDM standard as four postscript documents.

Duke University, North Carolina, maintains a repository for MSDS electronic files at:

(WWW)	http://dumccss.mc.duke.edu/ftp/standards.html
(FTP)	dumccss.mc.duke.edu

In addition, DICOM maintains electronic information at:

(FTP) xray.hmc.psu.edu

4.9.2 Health Level Seven (HL7) - Background

HL7 was founded in 1987 to develop standards for the electronic interchange of clinical, financial and administrative information among independent health care oriented computer systems; e.g., hospital information systems, clinical laboratory systems, enterprise systems and pharmacy systems. Currently, HL7 does not support images but is working with the ACR to merge the DICOM standard with HL7 for image support.

In the last three years, its membership has tripled to over 1,400 hospital, professional society, health care industry, and individual members including almost all of the major health care systems consultants and vendors. The HL7 standard is supported by most system vendors and used in the majority of large U.S. hospitals today. It is also used in Australia, Austria, Germany, Holland, Israel, Japan, New Zealand and the United Kingdom.

HL7 minutes, standard drafts, and sample source-code are available through Internet FTP servers on [dumccss.mc.duke.edu], WWW URL:

http://dumccss.mc.duke.edu/ftp/standards.html

Also supported is a discussion group on the HL7@Virginia.EDU list server.

Virtually all HIS vendors are HL7-compliant and most of the world, including the military, is merging their HIS systems and sub modules into this standard. However, each vendor's implementation of HL7 is somewhat different--a unique interpretation. Thus, while HL7 provides a strong measure of order to the messaging dilemma between HIS systems and sub-modules, it doesn't eradicate all communication problems. Interfacing two HL7-compliant systems, for example, requires much work on a technical level.

4.10 Data Interface Engines

Because of the complexity involved in interfacing modules to HIS systems, each with its own interpretation of a recognized messaging standard, many system integrators are turning to "data interface engines" to simplify the process.

Interface engines (IEs) are a complex middleware technology also known as integration engines, interface hubs, and application interface gateways. Typically, an IE is a separate computer which acts to translate and map data between other computer systems and their applications. These disparate applications must have the ability to exchange messages, for example through a messaging Application Programming Interface (API).

In the hospital environment, such IEs are used between HIS modules (i.e., the ADT module and the Radiology Module) perhaps purchased through different vendors with different hardware/software implementations. The benefits of using an IE include, 1) simplified HIS interface development since the IE is a tool-set designed specifically for that purpose, 2) centralized interface management capabilities (i.e., starting, stopping, monitoring, trouble-shooting), 3) superiority over point-to-point (PTP) interfaces since complexity is reduced through use of the centralized IE hub (i.e., if 5 different systems requiring bilateral interfaces need to interoperate with each other, 20 PTPs are needed, while only 10 interfaces are needed to an IE-based implementation--adding another node to the former requires 10 more PTPs while the later only 2 interfaces), 4) possible reduced costs for IE-based interface implementation when compared to

paying application vendors for installing a PTP-based interface, 5) the ability to populate clinical data repositories or data warehouses by routing data from messages exchanged between other applications, 6) an established CHIN entry-point for an organization.

IEs ideally send messages following the HL7 standards. However, some Electronic Data Interchange (EDI) transaction sets, and American Society for Testing and Materials (ASTM) messaging standards are also used.

4.11 HIS Networks and Standards

Many HIS systems connect various computer systems together within the hospital and these systems branch out to terminals for end-users. Such networks in the local environment are known as Local Area Networks (LAN). However, linkages to the HIS are not limited to within the LAN. External forces are pushing the inter-networking boundaries of the HIS.

It has become difficult for hospitals to stand alone. Health care reform is driving a new health care model-- a hospital today is just one stop along an entire continuum of care that can include other providers such as physician offices, home health agencies, Preferred Provider Organizations (PPOs) and Health Maintenance Organizations (HMOs). Local medical centers are joining together to become regional systems who are themselves tapping into national data resources to improve decision making and compare their performance to others nationwide.

Organizations must share caregiver information as patients move along the continuum. They must establish two-way links with national and regional data-bases to report and use ubiquitous data critical to ascertaining risk and providing cost-effective care. As a result, today's health delivery model is three-tiered, its orientation radiating outward from the local, stand-alone organization to the regional, community-based system to the national governing organization.

The following section examines some of the network technology being used to establish these Local Area Networks (LANs), Metropolitan Area Networks (MANs), and Wide Area Networks (WANs).

4.11.1 Ethernet, A Local Area Network Technology

Ethernet is a LAN technology that transmits information between computers at speeds of 10 and 100 million bits per second (Mbps). A LAN is defined as a privately owned data communications system that usually covers a relatively limited territory, hence the term "local area."

There are several LAN technologies in use today, but Ethernet is by far the most popular. Networking vendors estimate that as of 1994 there were nearly 40 million Ethernet nodes installed worldwide. The widespread popularity of Ethernet ensures that there is a large market for Ethernet equipment, which helps keep the technology competitively priced.

Currently the most widely used version of Ethernet technology is the 10-Mbps twisted-pair variety. The 10-Mbps Ethernet varieties include the original thick coaxial system, as well as thin coaxial, twisted-pair, and fiber optic systems. The most recent Ethernet standard is the 100-Mbps system which is based on twisted-pair and fiber optic media.

The ability to link a wide range of computers using a vendor-neutral network technology is an essential feature. Most LANs today support a wide variety of computers purchased from different vendors and require a high degree of network interoperability, which Ethernet provides.

For more information on Ethernet, see the on-line quick reference book, by Charles Spurgeon, through the WWW URL:

http://wwwhost.ots.utexas.edu/ethernet/descript-10quickref.html

4.11.2 ISO's OSI Model

ISO's OSI (Open Systems Interconnection) has established a non-proprietary communication reference model, split into seven layers, from physical cable definitions up to distributed databases and information systems, together with management and security tools. OSI data is available at the following URLs:

{http://cio.cisco.com/warp/public/535/2.html}
{http://www.adc.com/~don/osi/osi_1.html}--very good,
use as Appendix *, OSI Reference Model
{http://www.adc.com/~don/tech.html#tut}

4.11.3 Asynchronous Transfer Mode (ATM) Networks

In some multi-hospital networks, ATM technology is being used as a basis for sharing information along the continuum of health care. ATM allows interoperability of information, regardless of the "end-system" or type of information. ATM is an "emerging technology" driven by international consensus, not by a single vendor' s view or strategy.

Historically, there have been separate methods used for the transmission of information among users on a LAN, versus "users" on the WAN. This situation has added to the complexity of networking as user' s needs for connectivity expand from the LAN to metropolitan (MAN), national, and finally world wide connectivity. ATM is a method of communication which can be used as the basis for both LAN and WAN technologies. It is felt that over time as ATM continues to be deployed, the line between local and wide networks will blur to form a seamless network based on one standard-ATM.

Today, in most instances, separate networks are used to carry voice, data and video information-mostly because these traffic types have different characteristics. For instance, data traffic tends to be "bursty"-not needing to communicate for an extended period of time and then needing to communicate large quantities of information as fast as possible. Voice and video, on the other hand, tend to be more even in the amount of information required-but are very sensitive to when and in what order the information arrives. With ATM, separate networks will not be required. ATM is the only standards-based technology which has been designed from the beginning to accommodate the simultaneous transmission of data, voice and video.

4.11.4 Fiber Distributed Data Interface (FDDI) Networks

The high bandwidth technology provided by fiber optics, opens up new opportunities for very high multiplexed data rates. Rapid advances in this field will provide not only higher data rates for LANs but for the largest networks including the Internet. Fiber Distributed Data Interface FDDI networks will be growing rapidly where current bandwidth is limiting system performance. Fiber is ideal for the distribution of pathology images where the large bit counts demand wide bandwidth for reasonable response times.

4.12 HIS Systems (Civilian)

According to Ms. Donna Palumbo, Marketing Support Specialist, of Keane, Incorporated--a firm which markets HIS systems--the top four HIS Vendors are ranked as follows:

1) HBO & Company (Atlanta, GA)
2) Shared Medical Systems Corporation (Melvern, PA)
3) Medical Information Technology, Inc. (Westwood, MA)
4) Keane, Inc. (Boston, MA)

The following is a brief overview of these firms.

4.12.1 HBO & Company (HBOC)

HBOC is a healthcare information solutions company that provides information systems and technology for the health enterprise--hospitals, integrated delivery networks and managed care organizations. HBOC claims to offer products and services to meet virtually every need that the enterprise has for information, whether patient care, clinical, financial, or strategic management.

HBOC markets local, metropolitan and wide area network services; HBOC's client/server-based Pathways 2000 suite of applications provide key elements for integrating and uniting providers across the continuum of care and establish the infrastructure necessary for a lifelong patient record. Its hospital-based STAR, Series and HealthQuest transaction systems and TRENDSTAR decision support system--along with the clinician-focused Pathways 2000 products--help improve the delivery of health services to an entire community. The Pathways 2000 resource scheduling and managed care solutions and QUANTUM enterprise information system support the critical business functions necessary to manage today's emerging health networks. In addition, agreements and alliances with business partners allow HBOC to offer a broad variety of complimentary applications and technology, such as physician practice management systems.

HBOC wraps these products with such services as planning, implementation and support, plus education and training. HBOC also offers a range of outsourcing services that includes strategic information systems planning, data center operations, receivables management, business office administration and major system conversions.

4.12.1.1 HBOC's Network Solutions

HBOC has noted that healthcare is drastically changing in the way it conducts its business. Fee-for-service is giving way to managed care and competition. Stand-alone hospitals are being incorporated into health enterprises. Wellness is being measured by outcomes rather than amounts of care and patient chart size by transmission time rather than page count.

With such change, HBOC is attempting to address the following information requirement issues: 1) How do organizations share information among the many new players in a managed care environment? 2) How do they provide meaningful information for universal access throughout the facility? 3) How do separate organizations exchange the information required for a true computer-based patient record? 4) And how does any healthcare entity avoid system obsolescence in a technological environment that's advancing exponentially? 5) How do organizations build an information infrastructure to support a rapidly and constantly changing environment?

HBOC has formed "HBO & Company's Connect Technology Group" (CTG) to address the aforementioned issues based upon the conviction that retrieving, integrating and presenting information from disparate sources to an expanding variety of users will become critical in the new world of healthcare--and that networks will make these tasks possible. CTG has more than 20 years of healthcare industry knowledge, more than 100 healthcare network installations, advanced networking expertise and "proven experience" in providing information.

4.12.2 Shared Medical Systems Corporation (SMS)

Unfortunately, SMS only sent to Kensal Corporation literature describing their SMS OPENLab, a client/server laboratory information system (LIS). Since an LIS is a subset of an HIS, a brief overview of SMS and their OPENLab system is presented.

Apparently, SMS has been in the healthcare information systems industry for 25 years.

4.12.2.1 Voice Recognition and Multimedia

SMS OPENLab supports voice recognition and multimedia technology. Examples of multimedia features include on-line Help, CD-ROM reference manuals, scanned images for user-tailored Help files, full motion video and potential links to hospital satellite connections for remote training sessions, documentaries and network-wide continuing education and training opportunities.

4.12.2.2 Encoding Enterprise Rules

OPENLab automates administrative tasks and exception alerts while eliminating redundancy. Operational and clinical rules capabilities are embedded into OPENLab. For example, users can set up results reporting based on criteria such as location, choice of print media, day of the week or time, to ensure that results are delivered to the appropriate clinicians immediately and in the format they desire.

4.12.2.3 Open Systems Approach

OPENLab is based on an open system approach, enabling users to choose the technology and operating system that best fits their needs. Users may use off-the-shelf software such as report writers, spreadsheets, databases and word processing applications. Optionally, an OPENLab system includes an HL7/ASTM compliant interface engine to optimize network and system communications. Further, full support of point-of-care testing devices, faxes, printers and pagers in physician offices is provided.

4.12.2.4 Ad-Hoc Reports

Users can define ad-hoc report formats which integrate data, text, and graphical representation of results. The need for ad-hoc reporting was underscored by SMS since the laboratory marketplace is constantly changing. "Microsoft Access" was cited as an example of a "canned-package" that combines the power of a relational database with an easy-to-use graphical report writer.

4.12.2.5 Augmentable On-Line Help

Context sensitive on-line help can be augmented to include standard operating procedures, scanned images, CD-ROM reference manuals, and multi-media capabilities with full motion video. SMS claims that "any number of third party packages" may be used to include text and graphics into the Help feature.

4.12.2.6 On-Line Screen Editing

Rather than contracting SMS to alter screens every time a change is needed, an on-line screen editor is available which enables a user to tailor screens to meet individual specifications, improve system flow, and user productivity. The reconfigurable features are: the prompt text, tabbing sequence between fields, and the layout of fields over one or more screens. Changes can be executed throughout the system without bring the OPENLab system down.

4.12.2.7 Flexible Human Interface

OPENLab is GUI-based, multitasking compliant, and has user-definable security levels. In addition to support of mice, track balls, keyboards, and "hot keys"--light pens and touch-screen data entry options are available. A common user-interface model may be applied over the client/server technology; however, entity-specific (client) tailoring is allowed for improved end-user throughput.

4.12.2.8 Platform and Network Hardware

PC, IBM RISC System/6000, Digital VAX/VMS, Alpha, HP, Ethernet LAN

4.12.3 Medical Information Technology, Inc. (MEDITECH)

MEDITECH is a software and service company who develops, installs, and supports information systems for health care organizations of all sizes. MEDITECH emphasizes their technical innovation such as the new Handheld Point of Care Computer, and their "enterprise-wide computerized patient records." MEDITECH offers perpetual license agreements, periodic enhancements, ongoing education, and free system upgrades so customers can migrate to new technologies as they develop.

MEDITECH has 700+ installations (as of 1994) worldwide, with a majority of the customers located in the United States, Canada, and the United Kingdom. MEDITECH has averaged more than 80 new customers annually during the past five years.

Led by CEO A. Neil Pappalardo, MEDITECH has 1,300+ health care technology professionals at its five office sites outside Boston, MA. The staff is characterized by low turnover and "uncommon commitment" to the company.

MEDITECH emphasizes a flexible, integrated approach to information systems which provide patient-based information, open systems connectivity, and easy to use decision support tools necessary for today's community health care enterprises.

Clients may build information networks comprised entirely of MEDITECH applications or combine MEDITECH's modules with other vendor's products in open networks.

MEDITECH claims to place "up-to-the-minute information in the hands of care providers throughout a health care network, regardless of whether those providers work in hospitals, clinics, nursing homes, physicians' offices, or patients' homes."

MEDITECH boasts a design principal which mandates that information systems be easy to use. One example they point to is their PCI (Patient Care Inquiry) product, used by many physicians, and can "literally be learned in five minutes."

MEDITECH is a "financially stable company dedicated exclusively to the health care market." They pursue steady, long term growth rather than rapid expansion.

4.12.4 Keane, Inc.

From Keane's Corporate Introduction (#040195C): "Keane, Inc. designs, develops and supports software for corporations and healthcare facilities. John F. Keane founded the company in 1965 as a sole proprietorship and in 1967 incorporated the company in Massachusetts. Keane has since grown into a $350 million company with over 4,000 business and technical professionals. Headquartered in Boston, Massachusetts, Keane provides services across a network of over 40 branch offices throughout the United States and Canada.

"Keane's initial corporate objectives were to assist companies in the design, development and implementation of computer systems and provide project management services to Fortune 1000 firms. Keane is now also well known for its project management methodology, Productivity Management and for the ability to complete even the most complex projects on time and within budget.

"Keane's mission is to help organizations leverage their software assets and resources to achieve their business objectives. Keane strives to build long-term, mutually beneficial relationships with its client companies by effectively addressing their software development needs. Keane's success in meeting their needs has enabled the company to derive more than 90% of its annual revenue from existing clients. It has also resulted in Keane being recognized as one of the best managed small companies in the United States by publications such as Business Week, Forbes, Financial World and Investors Business Daily.

"Keane has two operating divisions: the Information Services Division (ISD) and the Healthcare Services Division (HSD). ISD provides custom applications software for corporations with large and recurring software development needs. Application software development includes systems planning, analysis, design, and maintenance. ISD also provides project management and help desk out-sourcing for clients.

"Keane's Healthcare Services Division develops and supports a full line of UNIX-based 'open' hospital applications including Patient Management, Financial Management, Patient Care and Clinical Systems. The Leadership Plus Series, a PC-based Long Term Care solution is Keane's offering for the long-term care market.

"Headquartered in Melville, New York, the Healthcare Services Division has branch offices in Hunt Valley, Maryland, and Los Angeles, California."

4.12.4.1 Healthcare Services Division Overview

From Keane's Division Overview (#070695C): "The Healthcare Services Division represents Keane's continuing commitment to bringing state-of-the-art application software and services to the healthcare industry. With technical, consulting, and management experience dating to 1975, Keane has grown to be a top provider of healthcare information systems in a very unstable marketplace.

"During Keane's early years, the healthcare unit experienced rapid growth by providing facilities management and assuming full responsibility for a hospital's information system needs, supplying the software, hardware, and the management and technical personnel needed to operate the hospital's information system.

"In 1984, Keane made its software available as a turnkey package. This full line of modular, yet integrated, software applications solidified Keane's reputation in the marketplace. In April of 1992, Keane acquired Ferranti Healthcare Systems Corporation, a software provider for acute-care hospitals and long-term care facilities. This acquisition expanded Keane's geographical presence in the acute and rehabilitation hospital market and added approximately 300 long-term care clients with 700 facilities located across the United States. In August of 1993, Keane acquired the software and selected assets of Professional Healthcare Systems, Inc. headquartered in Los Angeles, California. This acquisition brought to Keane a prestigious client base, including large teaching hospitals and several large healthcare chains. In April of 1995, Keane acquired the Infostat division of Community Healthcare Computing, positioning Keane among the top healthcare information systems vendors in the country and increasing Keane's install base to over 230 hospitals.

"Keane currently markets and supports a full line of information systems for the healthcare environment:

Threshold: A comprehensive hospital information system, uses open system computing technologies that combine RISC-based hardware, the UNIX operating system, a fourth generation programming language, and a relational database management system.

Patcom: a proven, highly rated Patient Management System designed for large teaching hospital and multi-entity facilities.

Leadership Plus: the premier financial and resident care system for long-term care facilities.

"In addition to application software, Keane offers support services that include new enhancements to meet changing regulatory requirements, hot-line, and remote diagnostics. Keane continues to offer both facilities management and transition management that provide either long-term or short-term on-site system support, training and management.

"Keane's current management team, with and average of twenty years of experience in healthcare information technology, has been working as a unit for more than fifteen years. Keane is known for its ability to provide a total solution including software, implementation, hardware, training, and documentation."

4.13 HIS Systems (Military)

The US military has standardized its HIS installations around the world through two systems: the Composite Health Care System (CHCS) developed by SAIC (Science Applications International Corporation) and the Decentralized Hospital Computer Program (DHCP) developed by the Veterans Administration (VA) Hospital.

4.13.1 Science Applications International Corporation (SAIC)

Science Applications International Corporation (SAIC), a privately owned defense contracting company headquartered in San Diego with 19,000+ employees nationwide, enjoyed $1.9 billion in revenues for FY94 with 86% coming from the federal government (USA Today, August 21, 1995). Outside the national security community, few have heard of SAIC.

Founded in 1969 by J. Robert Beyster, SAIC's principal product is brainpower. It acts as a systems integrator to design solutions to the government's toughest technology problems. SAIC's past projects include designing one of the early "star wars" antimissile defenses, the FBI's computerized fingerprint-identification system, and plant monitoring equipment for power plants. SAIC has also designed and built the largest hospital information system in the world as well as the largest medical telecommunications system in the United States (SAIC 1995 Annual Report). Recently, in a telemedicine experiment, SAIC helped link physicians aboard a hospital ship off the coast of Haiti with major U.S. military hospitals. As a result, the ship's doctors were able to give U.S. soldiers better medical treatment. Finally, SAIC's newest DoD health care contract involves building community networks linking military medical facilities with civilian providers and VA medical centers.

4.13.1.1 SAIC's Composite Health Care System (CHCS) Program

The Composite Health Care System is funded through a $1.1 billion contract, SAIC's largest program. CHCS fundamentally is an automated network handling military health care, including patient scheduling, admissions, prescriptions, lab tests, and record keeping and was

developed from close cooperation between the Pentagon and SAIC. CHCS has been installed in over 600 medical facilities worldwide and is also used in mobile units such as the one deployed in Gutanamo Bay, Cuba (discussed later in section 6.1.3).

4.13.1.2 The Mission of the Department of Defense in Health Care

The mission of the Department of Defense (DoD) health care system includes maintaining the health status of the military force (including family members and retirees and their family members) by providing cost-effective, high quality inpatient and outpatient medical and dental care and maintaining medical readiness to support mobilization. It includes all inpatient medical facilities and all significant outpatient facilities, to include care delivery in the military theater and veterinary services.

Medical data processing capabilities are being acquired to assist the health care providers and administrators in the management and delivery of quality care to the patient population served within the DoD health care system. A flexible solution is being provided in medical data processing capabilities for all DoD medical treatment facilities (MTFs). Both large and small MTFs will be supported via a standard Composite Health Care System (CHCS). The architecture design involves an integrated hardware and software solution, fully scaleable to the range of DoD medical facilities, from small stand-alone facilities to large regions and outpatient catchment areas (OCAs).

4.13.1.3 CHCS and MEDSITE (MEDical Systems Implementation and Training)

Approved by the Air Force Surgeon General in March 1993, MEDSITE's mission was to deploy CHCS to those Medical Treatment Facilities (MTFs) which had existing Initial Operating Capability (IOC) systems (TRIPHARM, TRIRAD, TRILAB, TRIPAS).

When PC-CHCS was approved for accelerated deployment to all other MTFs, Lt. Gen. Sloan approved a ramp up of MEDSITE and Standard Systems Center (SSC/SBM) to deploy CHCS Patient Appointing and Scheduling (PAS), Patient Administration (PAD), Managed Care Program (MCP) and Pharmacy (PHR).

SSC/SBM hired 4 of 38 needed term employees to deploy PC-CHCS to 29 MTFs in eastern CONUS/USAFE. MEDSITE hired 54 term employees to deploy PC-CHCS to 30 MTFs in western CONUS and PACAF, to manage the PC-CHCS project and to operate an AF CHCS Support Center.

MEDSITE currently maintains a software team which develops interfaces between CHCS and other various medical information systems, as well as report generators and other specific modules. Some interfaces are developed as a final deployable product while others are developed as a prototype effort to provide a proof of concept and provide a better understanding of the level of effort required to develop a fully functional interface for the system in question. The team also develops hard coded report modules in situations where using a generic ad hoc report generating tool is ill suited to the task either because of complexity or performance.

MEDSITE maintains WWW pages at URL: http://bender.brooks.af.mil/

This server has descriptions and M source code of the public domain software that is currently available from MEDSITE, and to be developed in the future. Some of the interfaces that have been developed are:

Telephone Refill
TransLux DataWall
Pyxis Medstation ADT

Provider Workstation Results Retrieval
TRAC2ES Patient Movement Request
MICROMEDEX

Some of the report generators that have been developed are:

Pharmacy Cost Reports
Medicare Eligible Cost Reports

MEDSITE's deployable systems have been installed at:

Guantanamo Bay, Cuba - (Operation Sea Signal)
Zagreb, Croatia - (UN Protective Forces)

MEDSITE's required future work includes:

Deploy CHCS LAB to all AF MTFs by Dec 95
Deploy CHCS RAD and Order Entry by Dec 96
Support training for software upgrades for existing MTFs

Future work for MEDSITE may also involve becoming or forming an executive agent for the Consolidated Medical Systems Support Center (COMSSC).

4.13.1.4 Case Study of Remote USAFB CHCS Site: Guantanamo Bay, Cuba

This section will provide excerpts from a May, 1995 USAF "After Action Report" which describes the humanitarian-mission/medical-effort carried out recently in Cuba, code-named "Operation Sea Signal". These excerpts will serve to explain how CHCS was deployed in a mobile context and what the various camp implementation issues were for that context.

Excerpts From Executive Summary of Operation Sea Signal

As part of Operation Sea Signal humanitarian mission, the Joint Task Force (JTF) 160 Surgeon General (SG) was responsible for the care and support of the 21,000 Cuban migrants and approximately 500 Haitian migrants housed at the Guantanamo (GTMO) Bay encampments. Specifically, the medical care for the migrants was provided by the 6th and 59th Air Transportable Hospitals (ATHs). There was a wide range of medical services provided by these ATHs.

There was little automation deployed with the 6th and 59th ATHs. The requirements for basic medical automation in an ATH are the same as any fixed medical treatment facility - pharmacy, lab, radiology, results retrieval, patient registration and electronic mail. The purpose of this deployment was to support these basic requirements as well as validate new requirements specific to a deployed unit.

The major deficit in GTMO and within the ATHs was the lack of any type of computer/communications infrastructure. Naval Base (NAVBAS) GTMO had a wide area network (WAN) but the ATHs were not located in any area easily linked to this WAN. Secondly, the telephone infrastructure was saturated. Within the ATH, administrative duties were accomplished through the use of personal laptop computers that people had brought from home stations. After 11 months of use, they were beginning to break down and there was much concern about replacements. Telephones were limited to "field" phones linked by 4-wire tactical lines. At the 6th, there was not any link to electronic mail within the ATH or a link into the Internet. At the 59th, located across the street from the Camp Bulkeley J-6 (USMC), they had found a means to link up to the J-6 Banyan Vines server through tactical wire to provide them with access to e-mail at home. The 59th had no connectivity within the ATH. The pharmacy at the 6th ATH had brought Z-248 Tri-Service Micro Pharmacy System (TMPS) but they continued to have

breakdowns. The 59th ATH did not have TMPS but did have the capability to use a personal computer (Z-248) with Pharmacy Label Producing Software (PHLAPS) for printing prepack labels.

MEDSITE's deployment of the Composite Health Care System (CHCS) to GTMO Bay was prompted by a request from the pharmacist assigned to the 6th ATH. After receiving approval from the ATH Commander, the JTF/SG, USACOM/SG, and the AF/SG, MEDSITE put together an DEC Alpha AXP capable of supporting a minimum of 25 concurrent users and enough disk storage for one year of on-line data. The system was installed in the 6th ATH with plans to tie all medical activities together.

Deployment Strategy/System Configuration

MEDSITE deployed a DEC Alpha (AXP) 3000/300 with CHCS Version 4.2/MU2 software. Peripheral hardware included DEC VT 320s, LA75 text printers, and Data South 300 XL label printers. Connectivity was via Local Area Transport (LAT) using DECServer 300s. Connectivity to outside locations was accomplished by connecting line drivers and bridges/routers through phone or tactical lines. Other specifics for hardware are listed below:

Product or Function	Item
CPU	DEC Alpha AXP 3000/300
	RISC based 125mHz
DEC	1.5 GB DAT backup storage tape
StorageWorks	1.2 GB Disk Drive
	CD-ROM
Memory	64 Megabytes RAM
Disk Storage	20 Gigabytes (10 - 2.01GB disk drives)
Backup	Disk to Tape
	Disk to Disk
Operating System	OpenVMS Version 6.1
Software	DSM Version 6.3d
	CHCS Version 4.2/MU2
Other	TGV Multinet
	PWS/TRAC2ES Interface software

Communications

The AXP only has a 10BaseT connector and the DECServer only has a 10Base2 connector. A Boca Hub with a 10BaseT and 10Base2 was used to connect the AXP with the DECServer 300s. Running 6-wire unshielded twisted pair within the 6th ATH, VTs and printers were connected to DECServer 300s.

A link between 59th to JTFJ6 already existed. The Camp Bulkeley J6 (USMC) had connectivity between their Banyan Vines server and the JTFJ6 Banyan Vines server in the Pink Palace. A tactical line from the 59th ATH had been run across the street to the Camp Bulkeley J6. Since the JTFJ6 at the Pink Palace was linked to the Internet, both the 59th and the Bulkeley J6 were linked to the internet. The goal was to link the 6th ATH into the same Banyan Vines server at the Pink Palace so we could access either the internet or the 59th ATH. If the NAVHOS (Naval Hospital) GTMO had access to the internet then we could theoretically access them once we were on the internet.

Linking the 6th ATH to the JTFJ6 Banyan Server. The Navy Communication Detachment (NAVCOMMDET) at GTMO provided two cable pair that we used to attach two AT&T 3510 line drivers and two DECrouter 90T1 bridge/routers. One end was attached to CHCS via the Boca Hub while the other router and modem were attached to the JTFJ6 Banyan Vine server. We had continuous problems with keeping the link up between the two modems. When the link was up we were able to telnet to the Banyan router and get to the internet.

Linking the 6th ATH to Camp Clinics (first is Lima/Mike camp) and the 59th to Camp Clinics (first is Echo/Foxtrot camp). Although two Codex 3500 line drivers were taken to connect Lima/Mike with the 6th, they were never tested because the lack of cable pair or commercial phone lines going to these clinics. A link in the future would require some type of wireless technology. LCDR Tillery and LT Welch visited from Naval Medical Information Management Center (NMIMC), they had discussed the installation of a cell on one of the hills and using cellular phones/modems to hook up the ATHs with their outlying clinics.

Connect to NAVHOS GTMO. The 6th ATH and the NAVHOS were both able to provide a single phone number that allowed modem access between the two facilities. Although not very fast we were able to link the NAVHOS Lab to CHCS using a pair of 2400 baud modems. Although we had taken 9600 baud modems we were unable to get the DECServer 300 to talk to them. Between the Pink Palace, Deer Point, and NAVHOS there is a clear line of site which is less than 4 miles total distance. Wireless technology could be used in the future.

4.13.1.5 CHCS Advantages

Results that were being recorded on separate log sheets and log books can now be found and printed in a collated report in less than five minutes compared to 20 minutes or more without CHCS. All specimens entered into CHCS were immediately accompanied by an audit trail providing positive specimen tracking. In addition special reports such as the Pending Lists, Overdue Procedure Reports, and Uncertified Results Report provided lab management an easy way to monitor the status of any test and take corrective actions to ensure results are returned in a expeditious manner not lost in a mountain of loose papers. Results were accessed from anywhere in the ATH there is a terminal, not just at the laboratory. This reduced the amount of time wasted walking to the lab to research what happened to a result. Electronic mail was used to pass information on protocol changes to different shifts, easing dissemination of critical operating policies.

4.13.1.6 CHCS In Emergency Unit

A CHCS terminal was placed in the Triage area (open tent adjacent to ER). This allowed the ER tech to triage the patient, take vitals, and print the 558 to the main ER. Changes were made to CHCS to allow the triage technician to enter directly into CHCS the patient's vital signs and to add comments he wanted to pass on to the ER. Once complete the 558 was printed on the ER printer. The bottom of the 558 was also changed to allow the understanding statement to print in Creole or Spanish.

An Information Desk Display was added to the Emergency Room main menu to allow for easy and fast look up of admitted patients.

4.13.1.7 DMPITS Database Conversion

Patient tracking was a problem at the ATHs. Some sections were using the US Atlantic Command (USACOM) developed Defense Mass Population Identification and Tracking System (DMPITS). There were multiple problems identified with DMPITS: (1) lack of confidence in the data accuracy because registration information was not verified at the time enrollment; (2) lack of

devices in each section (many of the devices were broken and did not work); and (3) the DMPITS was not on a network, leaving each section to build their database. DMPITS was updated manually once per week based on data provided to a central location. CHCS would provide a means for accurately tracking patients through the ATH as all would use one central patient database.

The DMPITS office provided a DOS "flat file" containing the Name, DMPITS Number, Date-of-birth, Sex, Camp, Tent, and Bed. This file was transferred to the Alpha using a laptop computer. A conversion program was written in MUMPS to read the file and insert the data elements into the CHCS database providing pre-registration for all migrants.

4.13.1.8 After Action Conclusions

The DEC Alpha proved to be the ideal platform for simplified system management required for a deployed system. A single CPU system eliminated the problems with database synchronization and greatly simplified back-up procedures. The performance was excellent and better than expected. Any deployable system should be fully scaleable if future upgrades become necessary. Finally, the OpenVMS operating system was very robust and tolerant of unexpected "crashes" that are often a fact of life when operating in a tent environment operating off generator power. All the ATH components (CPU, DECServers, VT 320s, LA 75s) were configured at MEDSITE and tested for compatibility and reliability prior to deployment. This part of the deployment went smoothly and as predicted. However, the remote communication solutions between all the medical facilities at GTMO were not tested because the availability of the type of physical wire was unknown. In the future one needs to know the location of the nearest Wide Area Network (WAN) connection and the locations of any remote sites that will be connected to the CPU. The distances from the CPU to these locations must be known as this will drive the communication solutions. Based on this information, the team should deploy with one or more solutions for each type of remote connections. The deployment to GTMO was very successful. The ability to get daily A&D reports; the ability to track the pregnant migrant women by camp, DMPITS number, and EDC; the ability to better track and monitor drug distribution, whether by prescription or by bulk issue; the ability to quickly send panic lab values directly to the clinic or ER; and the ability to register and track patients all improved the efficiency and quality of care being given by the 6th ATH. From this test deployment, many lessons were learned regarding the flexibility of CHCS and the flexibility required to support both humanitarian as well as wartime missions. These lessons will be used to better train our people for future deployments.

4.13.2 Military and HL7

Both the CHCS and the DHCP are based on the File Manager, a set of extensions to MUMPS originally developed by the VA that facilitates data sharing among applications that are homogeneously developed using the File Manager toolset. The coupling among FILEMAN applications is very tight, being based on a shared database. The development paths have diverged, however, so that the systems are not at all interoperable.

The DoD agency that issued the contract to SAIC published a notification in the Federal Register sometime in 1993 of its intent to use HL7 in the CHCS for interfaces to third-party systems.

The VA is adapting as rapidly as it can to HL7. Its people participate in the meetings and it uses HL7 for exchange between the DHCP and third-party systems. It has even looked at HL7 as a model for inter-module data exchange within the DHCP.

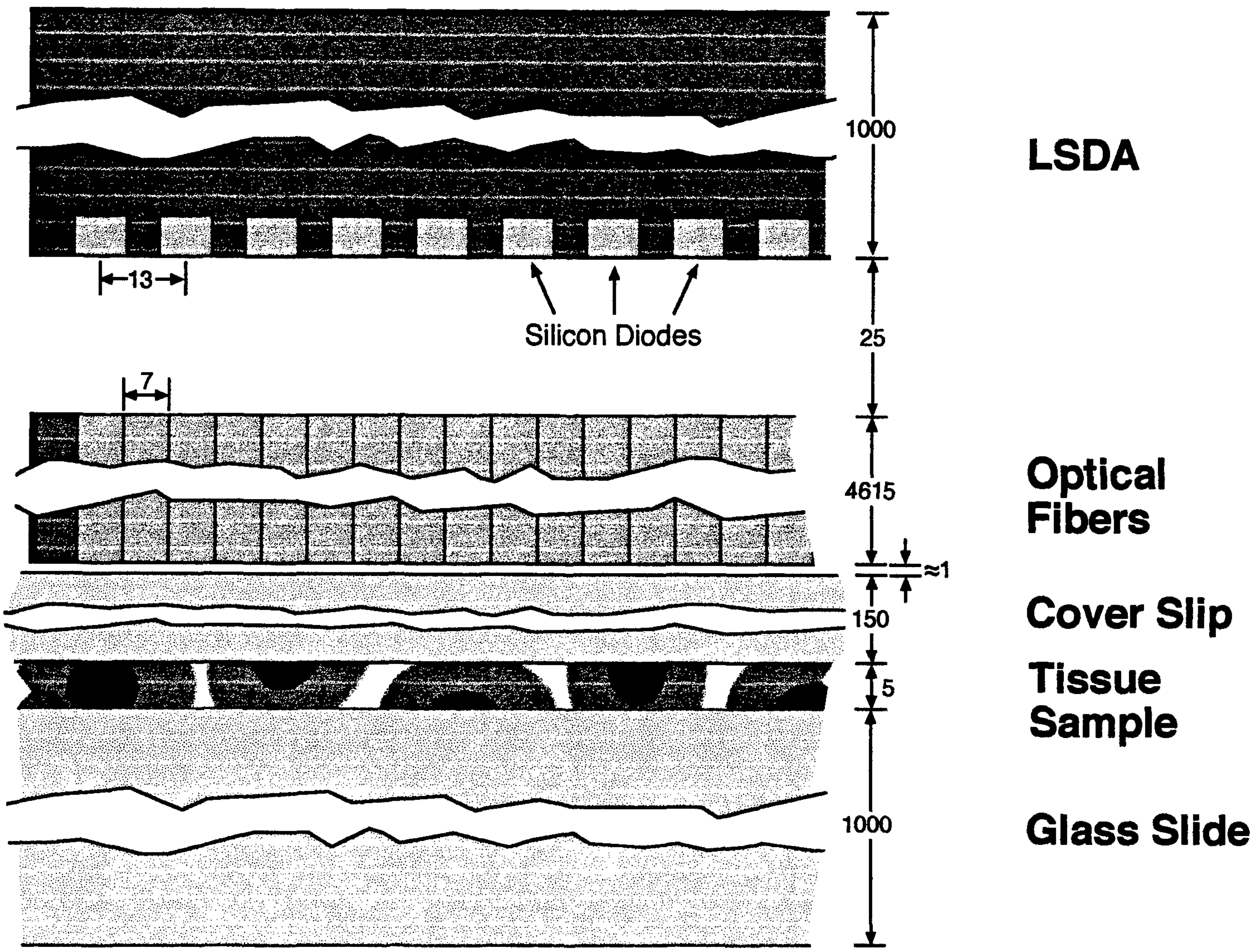


Fig. 8 - Schematic of the lensless microscope (numbers are in micrometers) .

5. SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS

Section 5.1below is quoted in part from Kensal's final report on the SBIR Phase I research project (DMI-9460231) for NSF (National Science Foundation). It provides a simple explanation of lensless microscopy - the key aspect of our research on computerized microscopes for military pathology. The objective of this research for the NSF was to investigate the possibility of quadrupling the number of picture elements per unit area in lensless microscopy. If successful this would increase resolution to the point that this new form of microscopy would be useful in picturing all types of human tissue.

First, let us explain how the lensless microscope works. The basic principle of the lensless microscope is illustrated in Figure 8. The specimen to be visualized by the microscope is a tissue section mounted in the traditional fashion on a microscope slide and covered by a thin glass coverslip. Light passes upwards through the microscope slide, through the tissue, and through the coverslip. In the traditional microscope an image of the tissue is formed with an objective lens/eyepiece combination on either the user's retina or on a picture-taking device such as a digital CCD television camera or photographic film camera. The deficiency with this traditional approach is that even using lenses with the largest field of view, only a small portion of the specimen can be visualized by the user. Even when using the lowest power microscope objective readily available, the user cannot view a large portion of the specimen. At lowest power, only 2% of the specimen is visible using a standard 640x480 TV camera. This increases to about 7% when using a 1000x1000 electronic imaging device.

5.1 Initial Invention

In the early 1980s the PI realized that, with advances in technology (fiber-optic couplers and linear CCD arrays with small spacings), an all solid-state microscope could be built that required no lenses and whose field of view would be unlimited. Figure 8 shows how, in such a microscope, light can be carried from the specimen, using optical fibers, to the light sensitive silicon diodes in a linear CCD array. After a patent was applied for by the PI in 1984, this idea of lensless microscopy was found to be novel and useful by the U.S. Patent Department. A patent was issued to the PI in 1988. In 1989 funds were first requested to make a demonstration by building a prototype of a lensless microscope. After many tries, a grant for a demonstration of lensless microscopy was made in 1992. Within five months, the world's first lensless microscope was built (Figure 9) and images generated. It is currently being applied to medical microscopy in the combined L/L (Lensless/Lensed) system that is under field trial in a telepathology hookup between the Mayo Clinic and Luke Air Force Base.

Using the apparatus shown in Figure 9, KSC offered to generate a lensless image of any microscope slide submitted to our laboratory. Our colleagues throughout the USA have taken advantage of this offer. Certain individuals have replied that they

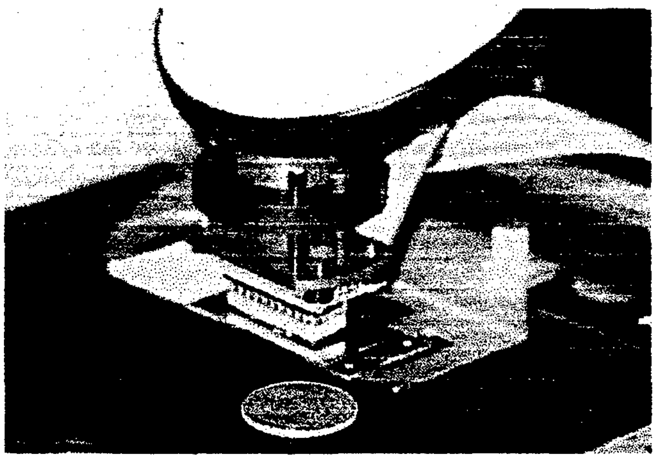


Fig. 9 - Kensal Corporation's laboratory demonstration of a lensless microscope.

can in fact make a diagnosis from the lensless image generated by our equipment. On the other hand, others have replied that the lensless image looks so "fuzzy" that even areas of interest could not be found. These reactions tend to be tissue specific. Sections of the human lymph node appear to present the greatest problem. Dr. Bharat Nathwani (University of Southern California and the Los Angeles County Hospital) told us that, in lensless images of the lymph node, it was sometimes impossible to select areas for examination at high magnification in that distinguishing features were missing.

5.2 Resolution Improvement

Therefore, support was requested from the National Science Foundation in 1994 to allow us to work with EG&G Reticon (Sunnyvale, CA) to devise a higher-resolution lensless microscope. We proposed that a tapered fiber-optic bundle be sliced and affixed to the original EG&G detector array to create a lensless magnifier as shown in Figure 10 having a 3.5 micrometer to 7.0 micrometer taper. The proposal was submitted in 1994 and funded in the fall of 1995. The budget allowed $13,900 for the entire diode array device. Due to several delays caused by EG&G Reticon, research was delayed and NSF extended the completion date of the Phase I research to September 30, 1996.

During this extended time frame EG&G announced a breakthrough, namely, a new product wherein the diodes themselves were spaced on seven micrometer centers. This explained why they had delayed in fabricating the diode array that used thirteen micrometer centers and the tapered fiber-optic faceplate. Unfortunately, the new diode array had an entirely different pinout. Also, instead of using two output pins for the alternating odd and even diode outputs from the linear array, a single output pin was employed with a time multiplexed odd/even output. Modifications to compensate were done by Kline Research (Reseda, California) based on parallel work being conducted for the Army Missile Command (DAAH01-95-C-R209). Within the original budget $13,900 Kensal was able to buy the new seven micrometer diode array, have new driver electronics fabricated and a new mechanical mount for the diode array fabricated (Boeckeler Instruments, Inc., Tucson, Arizona).

Due to the time required for this major realignment of our research program, the first demonstration of lensless imaging using seven micrometer diodes was not made until the end of August 1996. The first seven micrometer images were generated at the Kline Research facilities and proved successful. Immediately the entire laboratory apparatus was transferred to our laboratory for further experiments in the last month of the grant. First, it was found that the stage velocity (controlled by a JM-1 Boeckeler Instruments Motion Controller) was misadjusted. Kensal

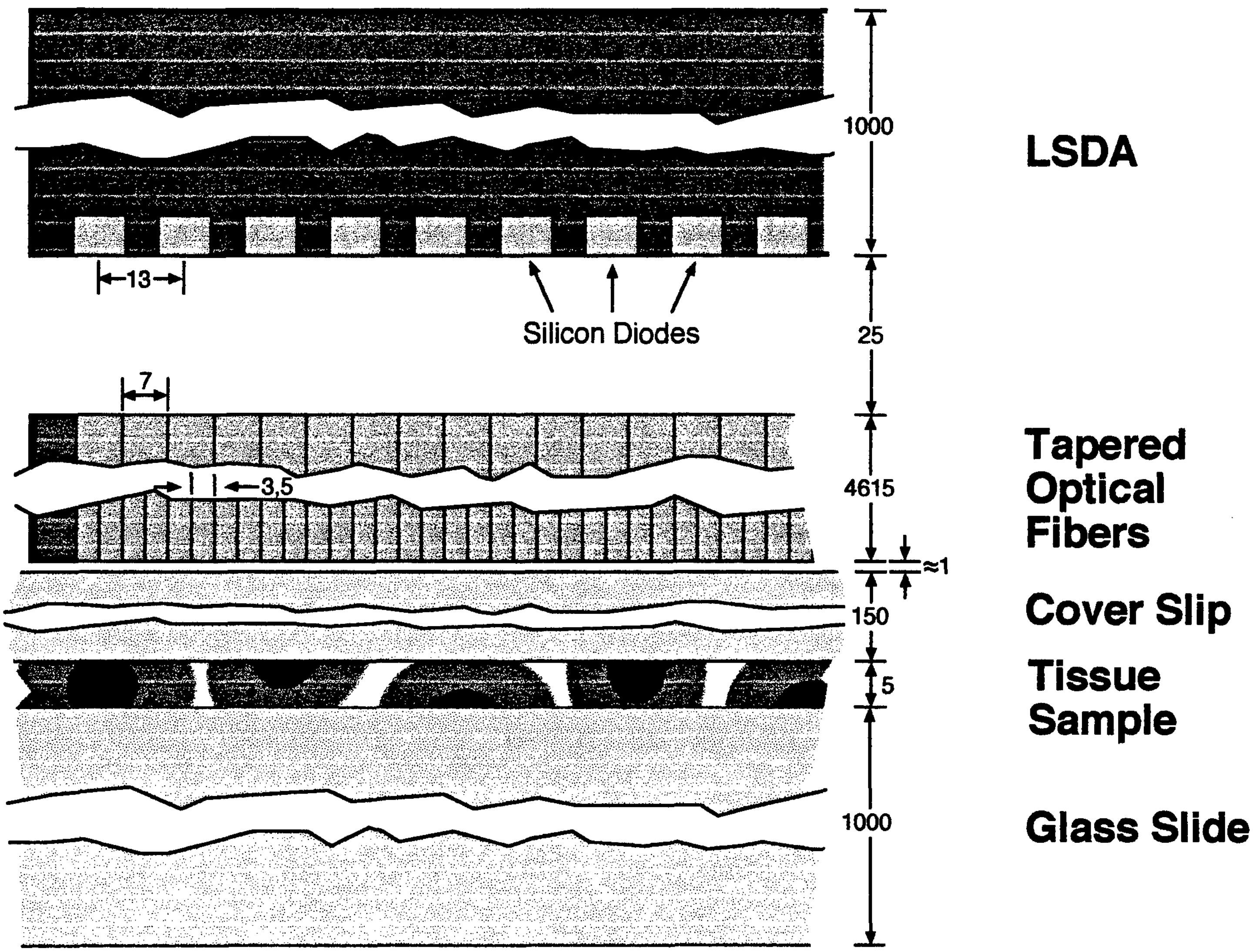


Fig. 10 - Lensless microscope with 2:1 fiber optic coupler.

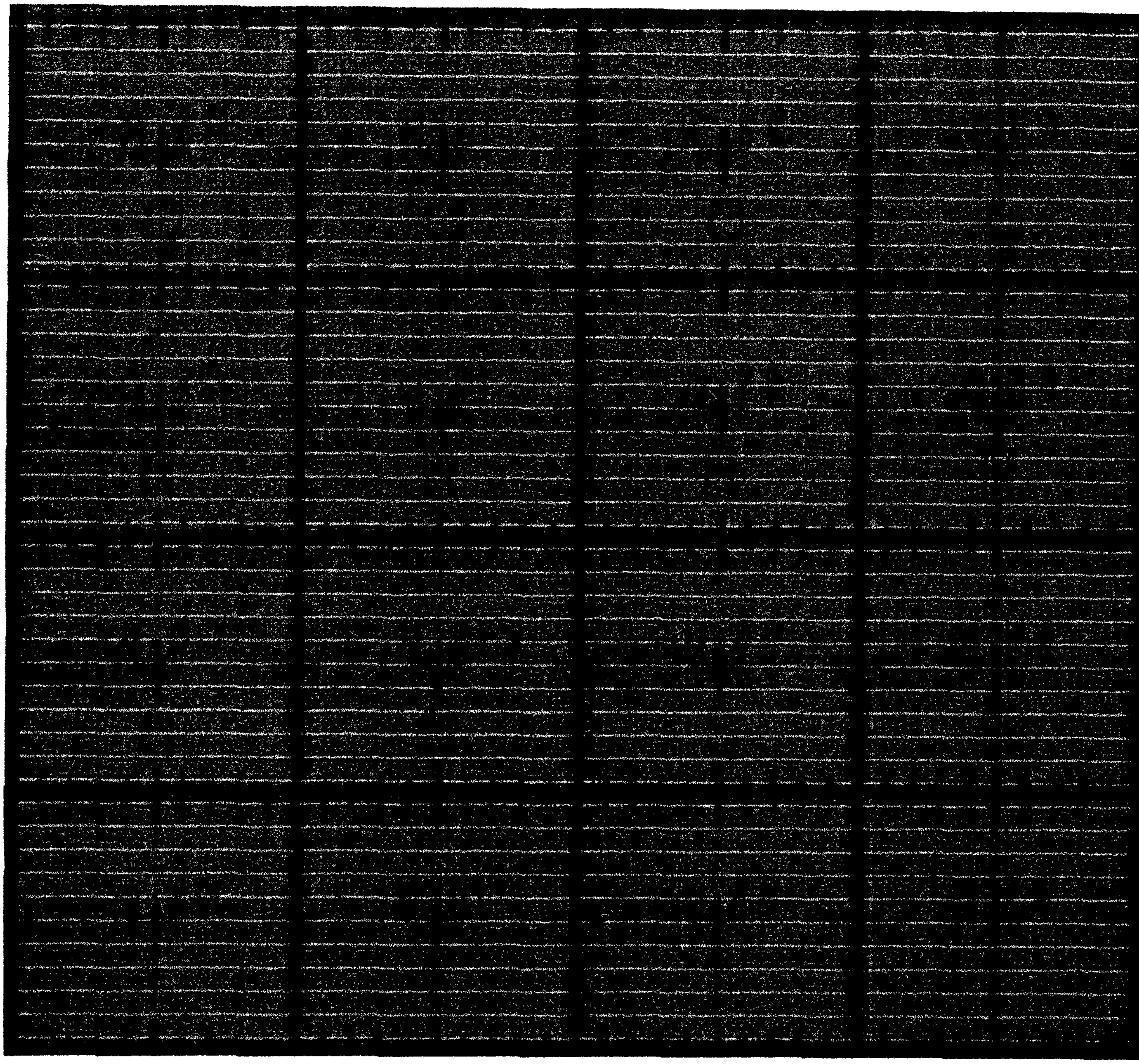


Fig. 11 - Lensless image of a 4x4 mm portion of a microscopic test pattern using a diode array scanner with 13 micrometer spacing.

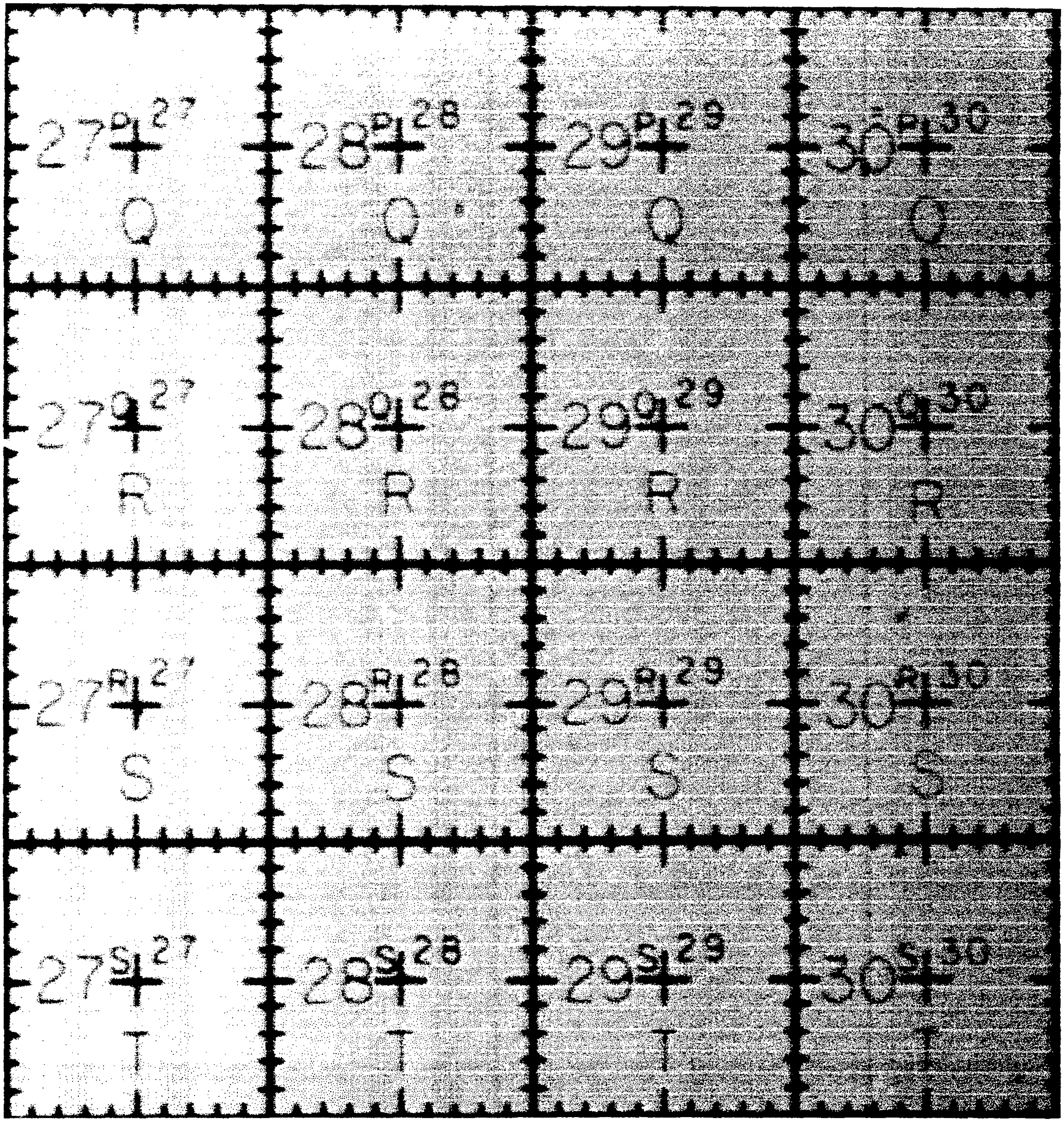


Fig. 12 - Same as Figure 11 except that the diode array uses diodes spaced at 6.9 micrometers.

staff worked on the software changes required to correct the motion contoller velocity and within a few days produced geometrically correct images. These showed remarkable improvement over results previously obtained from the thirteen micrometer diode array procured under our NIH grant. This advance fully satisfied the goals of the Phase I research for NSF. For example, Figure 11 is a lensless image of a matrix of 4x4 on millimeter squares taken with the original 13um scanner in 1992. Figure 12 is a lensless image of the same matrix using the new 7um scanner that was delivered to us in August 1996. The resolution improvement is obvious and dramatic. Numbers and letters that were scarcely visible in Figure 11 are clearly distinguishable in Figure 12. Thus this represents a highly important improvement in lensless scanning. The images are no longer "fuzzy" and should be interpretable by any pathologist.

5.3 Reprogramming our Research and Rebudgeting

As soon as our success on the NSF Phase I research occurred, we contacted the Boeckeler Instruments company that had participated in the two L/L (Lensless/Lensed) workstations now deployed at Luke Air Force Base and Mayo Clinic. Boeckeler was asked to quote on a retrofit for both workstations; one with just the new diode array and the other with the diode array plus a new stage and microscope so as to prepare for the fact that microscope models in use at Mayo Clinic and Luke AFB are being discontinued. This would yield two retrofitted workstations with high-resolution (6.9 micrometer) scanners that should revolutionize their performance. The PI feels that this is essential to the success of continued research in military telepathology for USAMRMC.

In order to perform the retrofit, we are recommending a rebudgeting according to the following table. This table shows the most recently approved budget and also the revised budget that transfers funds from the tasks involving field trials in Texas (see below) to tasks involving retrofitting the two existing workstations.

		Year 2 Budget		Year 3	Year 4
		Budget (approved 12/95)	Spent	Proposed	Proposed 6 mo. Extension
1.	SALARIES (W-2 and 1099) [1]	160,000	98,715	157,562	59,581
2.	BENEFITS [1]	--	--	--	--
3.	CONSULTANTS				
	Nance [2]	27,000	--	27,000	--
	Devey [3]	--	3,667	6,000	--
	Kline [2]	--	1,800	--	--
	Deasey [2]	--	510	--	--
	Garrett [4]	--	--	14,758	--
4.	EQUIPMENT				
	PCM Assemblies, Optics, Workstations	88,804	--	--	--
	Upgrade 2 existing workstations [2]	--	--	178,673	--
5.	SUPPLIES & MATERIALS	11,281	15,975	27,218	--
6.	TRAVEL				
	PHX-SFO	1,832	--	--	--
	PHX-SAN	1,408	868	--	--
	Within Arizona	--	1,576	502	--
	TUC-LAX	--	1,058	1,500	--
	PHX-Wash. DC [3]	--	--	3,500	3,000
7.	ADMINISTRATIVE SUPPORT	--	--	--	--
8.	INDIRECT COSTS	32,629	42,904	79,492	18,525
9.	MISCELLANEOUS				
	Loral	26,300	--	--	--
	Optical Systems Corp. [5]	50,000	69,800	35,053	--
	Washington, DC location	--	--	20,000	--
10.	TOTAL COST	399,254	236,873	551,258	81,106
11.	Obligated as of 9/30/96				
	Contract Labor	--	33,800		
	Subcontract [5]	--	111,600		
12.	TOTAL COSTS PLUS OBLIGATIONS		382,273		

Notes:

[1] Kensal in some cases now supports medical insurance for certain of its employees. When this is done, costs will be taken from salaries.

[2] Proposed use of funds originally earmarked for Loral in FY1997 to upgrade software (Nance), electronics (Kline), optomechanics (Boeckeler) of existing workstations.

[3] Coordination of workstation liaison in Washington, DC.

[4] Coordination of workstation effort at Mayo Clinic and Luke AFB.

[5] Completion of PCM prototype initially contracted in FY1995.

It is clear both from the images generated and the technical characteristics displayed in the table above that a significant improvement in L/L Microscopy for military medicine would take place if the retrofit is implemented. Therefore, it is the Kensal Corporation's strong recommendation that this retrofit be undertaken at the earliest possible date. The budget presented above will make this possible and could be started immediately when that approval has been received by USAMRMC.

5.4 Reorganization of Field Trials

Also, in order to satisfy ARPA's desire for field trials in Washington - not Texas, this has also been addressed. Per the recommendation of Dr. Richard Satava (head of the Advanced Biotechnology Program at ARPA) we are planning to arrange a transfer of the Mayo workstation from Scottsdale, Arizona, to one of the installations with whom we have been working in the Washington area. One candidate, recommended by Dr. Satava, is AFIP (Air Force Institute of Pathology). Another location where there is significant interest in telepathology is the Department of Pathology, School of Medicine, Georgetown University. Dr. Norio Azumi as well as Ms. Yukako Yagi have been working with us on comparing the partial coverslip scanner of Polaroid with the full coverslip scanner that has been developed under our NSF grant. A cross comparison of the characteristics of these two systems is given in Figures 13 and 14. As can be seen, at least for these images, the coverage obtained by the scanner at Georgetown is smaller than that obtained by our scanner. Resolution has been measured, but high magnification images have yet to be generated. The scanner at Georgetown generates about 12,000 picture points per square millimeter; the new lensless scanner, 21,000, i.e., an 80% increase in resolution.

Thus, if posssible field trials will be arranged at both locations.

5.5 Positive and Negative Aspects of Grant Research in FY 1996

The USAMRMC requests that each annual report summarize the year's research by giving both positive and negative aspects of the research. There are described here.

5.5.1 Positive Aspects

The major positive aspect of our Year 2 research was the acceptance and certification of the two workstations for use in telepathology that we contracted to be built in Year 1 by Boeckeler Instruments Inc. These two workstations are now the <u>only two in the world</u> to combine lensless and lensed microscopy in an integrated unit for both local and remote examination of surgical specimens of human tissue. Using a workstation, both local and remote examinations may be done by the user from images displayed on a high-resolution computer screen. Remote examination requires, in addition, the use of the ISDN (Integrated Services Digital network) for image transmission. Since the second quarter of FY1996 both workstations have been deployed: one at the Mayo Clinic; the other at Luke Air force Base. To date 40 surgical specimens have been analyzed by a team of four pathologists using both local and remote techniques. The study is double blind so that, after completion, a statistical analysis may be performed (FY 1997) to determine and compare diagnostic success rates using the digital imaging workstation versus using the ordinary manual microscope.

5.5.2 Negative Aspects

First, the Boeckeler workstation installed at the Mayo Clinic has shown satisticatory performance, the one at Luke AFB has not (Section 2.2). As of mid October the Luke AFB workstation has been returned to Boeckeler for overhaul.

Second, Optical Systems Corp. has been unable to deliver PCM (PC Microscope - a significantly more compact workstation for medical microscopy) on time. Thus we have cancelled the "production order" for any other PCMs and are requesting the funds be transferred to the now far more important research in retrofitting the existing Boeckeler workstations with high resolution lensless scanners. OSC is still funded to build a single PCM prototype using funds committed to them in FY 1996.

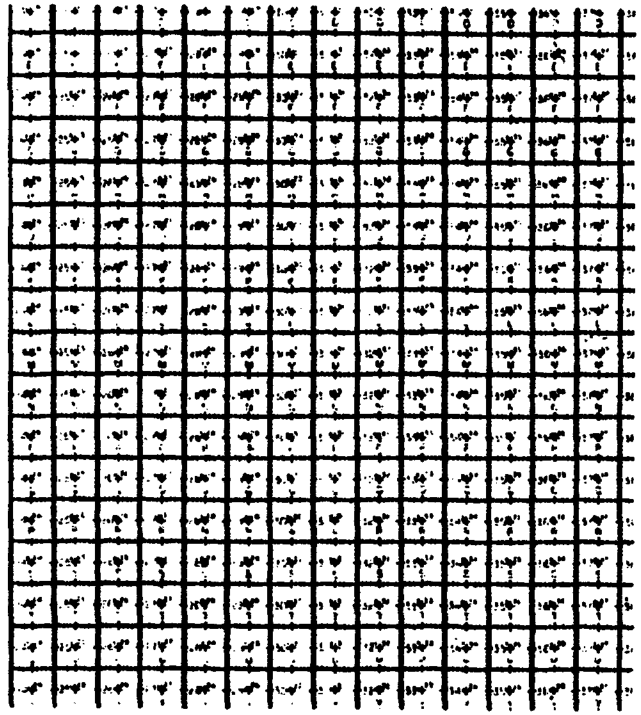


Fig. 13 - Lensed scan of test pattern microscope slide (23x44 mm) at Georgetown University. Coverage is 14.4x17.7 mm.

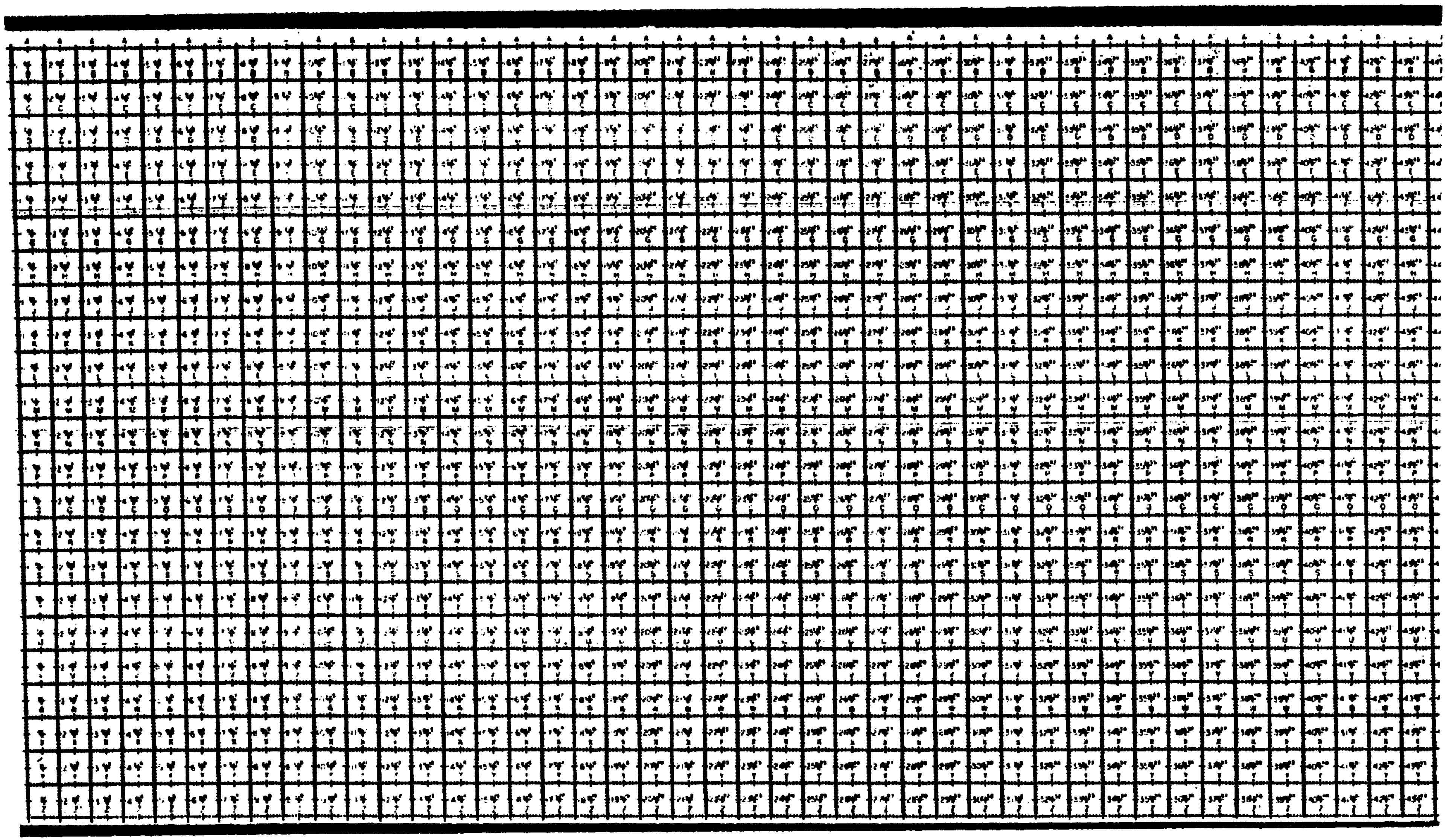


Fig. 14 - Lensless scan of 23x44 mm test pattern on microscope slides. Coverage is 100%.

Third, Loral, a company that showed interest in working with us on incorporating our L/L Microscope with their MIDIS (Medical Image Display System) have been merged into Lockheed Martin. Since them two telephone calls and one letter have brought no response. We feel that they must be dropped from our USAMRMC project and, as with OSC's research on PCM, funds transferred to reworking the existing workstations built by Boeckeler. The strategy will be explained more fully in a separate letter to USAMRMC.

APPENDIX A

1996 HIMSS/HP LEADERSHIP SURVEY

1996 HIMSS/HP LEADERSHIP SURVEY

For better and for worse, managed care is driving health care automation. Managed care continues to be the dominant force in health care as organizations focus on the bottom line and control costs. This is according to the seventh annual HIMSS/Hewlett-Packard survey of trends in health care computing which represents the opinions of more than 1,200 participants at the 1996 Annual HIMSS Conference in Atlanta, GA. Following is a summary of major findings.

49 percent of respondents believe the need to control costs due to the continued pressures of managed care is the most significant force driving IT investments in health care. Cost control outranked other forces such as competition with other providers (20 percent) and coping with mergers and acquisitions (17 percent).

Respondents were divided over the real-life impact of managed care. While slightly more than half (57 percent) say managed care will have a positive impact on health care by either lowering costs through consolidation or improving outcomes because of heightened focus on measuring quality, a substantial minority (41percent) are worried about the negative consequences of managed care. Among all respondents, 26 percent say that business forces will negatively impact clinical practices, and 15 percent believe that patient mistrust of gatekeeper physicians will grow.

Greatest IS priorities: upgrade, integrate and re-engineer

The most important IS priorities for health care organizations are upgrading their IT infrastructure (32 percent) and integrating systems in a multivendor environment (27 percent). Reengineering to a patient-centered computing environment is also receiving priority attention from 23 percent of health care organizations. And organizations are following through by completing these projects. Forty percent of organizations have undertaken projects to upgrade their IT infrastructures and 18 percent have begun systems integration projects this past year.

Strong movement beyond hospitals' walls

Reflecting the larger trend in health care delivery, computer technology is moving rapidly beyond the walls of traditional hospitals. The two greatest departmental automation priorities for the coming year are physicians' offices (35 percent) and outpatient clinics (15 percent), far outpacing traditional inpatient settings such as critical care, OR and Med/Surgery.

Computer technology in an office or group practice setting

In the outpatient setting, the greatest advantage of information technology, according to survey respondents, is access to current patient information across the enterprise (45 percent). Other advantages cited are: automating workflow (19percent); better financial management of offices (16 percent); and better management of non-clinical patient tasks (13 percent).

IS frustrations: where's the strategy?

Three out of 10 respondents (31percent) say their organizations lack overall strategic IS plans and are too focused on tactical projects. This represents an even higher degree of strategic frustration than one year ago, when 19 percent of respondents gave this response.

Emphasizing this further, the need to develop a strategic plan was cited as the greatest telecommunications challenge by 25 percent of respondents. Other frustrations include a lack of

applications to meet the demands of clinical data repository and/or electronic medical records (18 percent) as well as difficulty in finding and maintaining a good technical staff (16 percent).

Budgets and staffing on the rise

Eight in ten respondents say their IT budgets will increase over the next two years. This year's survey indicates a small drop in significant budget increases (defined as 30 percent or more budget increase), perhaps reflecting a "coming-to-terms" with the realities of the economic constraints of managed care. In a related but somewhat surprising finding, six out of 10 respondents believe their M.IS staff will increase over the next two years. This may indicate a realignment in response to previous downsizing trends.

The Internet and The World Wide Web in health care

The revolution in cyberspace has reached health care. The HIMSS/HP survey indicates that the most common use of the Internet is for on-line clinical research services &shyp; say 56 percent of respondents &shyp; and for physician-to physician communication, say 33 percent of respondents. Health care organizations also see the promise of the World Wide Web, the multimedia section of the Internet. Thirty-six percent of respondents say their organizations have Web sites up and running; another 37 percent are currently developing sites.

CHIN fever subsiding?

Nearly three-quarters of respondents (73 percent) say their organizations do not belong to a community health information network.

Integrated delivery systems (IDS)

Sixty-four percent of respondents say they are either part of an IDS or are in the process of becoming part of an IDS. Nineteen percent are still not part of an IDS but plan to become part of one within the next year. However, these findings reflect no change in the percentages from last year's survey.

CPRs &shyp; still weighing the options

Despite significant interest in computer-based patient records, nearly six out of 10 respondents say they have made no investment or committed to funding a CPR project. In contrast, about 30 percent of respondents say they have invested heavily in the equipment and software needed to implement a CPR.

Data storage

A significant majority of respondents see an explosion in data storage requirements over the next two years: 65 percent say storage requirements will more than double, another 20 percent predict the need for at least double their current requirements, and 13 percent say the load will increase by.about half of their current requirements.

Security of medical information

The bad news: eight out of 10 respondents (79 percent) are concerned about unauthorized access to computerized medical information. The good news, however, is that half of those concerned have taken steps to protect their data and the other half are planning to do so.

Data on telemedicine

Forty-one percent or respondents are at least somewhat involved in telemedicine, and another 35 percent are investigating it actively.

Praise for telecommunications reform

Sixty-four percent believe the new telecommunications law deregulating the industry will have a positive impact on health care, offering the potential of making telemedicine and videoconferencing easier than ever before. Only one in ten believe that it will have a negative impact.

Futuristic health care technologies

Half of the survey respondents agree that in the next three years access to on-line health care information and services from the home will be the most significant health care-related computer development affecting the average consumer (48 percent).

APPENDIX B

GLOSSARY OF ACRONYMS

Glossary of Telemedicine and Hospital Information Related Systems Acronyms; 12/10/98

No.	Acronym	Definition
1	AAMT	American Association for Medical Transcription
2	AAPA	American Association of Pathologists' Assistants
3	AARP	American Association of Retired People
4	ABI	Application Binary Interface
5	ABP	American Board of Pathology
6	ACGIH	American Council of Government and Industrial Hygienists
7	ACH	Automatic Clearing House
8	ACR	American College of Radiology
9	ADA	American Dental Association
10	ADC	, Analogue to Digital Converter
11	ADT	Admission Discharge Transfers
12	AHIMA	American Health Information Management Association
13	ALA	American Library Association
14	AMA	American Medical Association
15	ANA	American Nurses Association
16	ANSI	American National Standards Institute
17	APG's	Ambulatory Patient Groups
18	API	Application Program Interface
19	APM	Anatomical Pathology Module, part of an HIS
20	APT	Anatomic Pathology Test?
21	AR	Accounts Receivable
22	ARPA	Advanced Research Projects Agency
23	ARUP	Associated Regional and University Pathologists
24	ASCII	American Standard for Code Information Interchange
25	ASCP	American Society of Clinical Pathologists
26	ASTM	American Society for Testing and Materials
27	ATA	American Telemedicine Association, 512-480-2247
28	ATIS	Alliance for Telecommunications Industry Solutions
29	ATM	Asynchronous Transfer Mode, Automatic Teller Machine, Adobe Type Manager
30	AUI	Attachment Unit Interface, Ethernet transceiver cable between actual interface (computer) and the MAU
31	B-Channel	Bearer channel, ISDN channel with 64 kbps bandwidth (see PRI)
32	B/AR	Billings, Accounts Payable
33	BAI	Basic Access Interface, ISDN with two B and one D channels (2-64kbps, 1- 16 kbps), (2B+D)
34	BISA	Biomedical Informatics Society of Argentina
35	BLOB	Binary Large Object
36	BNC	A common type of quarter twist connector for coaxial cable.
37	BRI	Basic Rate Interface-16kbps ISDN Channel
38	BSF	Blood Systems Foundation, Scottsdale, AZ
39	BWH	Brigham and Women's Hospital
40	CA	Cancer Antigen
41	CAP	College of American Pathologists, Central Arizona Project
42	CBER	Center for Biologics Evaluation and Research
43	CBS	Common Basic Specification; Columbia Broadcasting Systems
44	CCBC	Council on Community Blood Centers
45	CCD	Charge Coupled Device, uses Photovoltaically generated packets of charge that are converted to pixels.
46	CCITT	Standards group now called ITU-T
47	CCOPE	Conjoint Committee On Pathology Enhancement
48	CDCP	Centers for Disease Control and Prevention
49	CDM	Common Data Model
50	CDR	Clinical Data Repository
51	CDRH	Center for Devices and Radiological Health
52	CEA	Carcino-Embroyonic Antigen
53	CEN	European Standards Group
54	CEN/TC 251	Working on spec similar to HL7. (CEN and HL7 coordinate)
55	CEO	Chief Executive Officer
56	CERC	Concurrent Engineering Research Center (CEEC) - WVU
57	CHCS	Composite Health Care System
58	CHI	Centre for Health Informatics, Wales
59	CHIME	College of Healthcare Information Management Executives
60	CHIN	Community Health Information Network
61	CHRP	Common Hardware Reference Platform, Multi-Platform operating system for computers
62	CIS	Clinical Information System
63	CLIA	Clinical Laboratory Improvement Amendment
64	CLIAC	Clinical Laboratory Improvement Advisory Committee
65	CLK	Clerk

Glossary of Telemedicine and Hospital Information Related Systems Acronyms; 12/10/98

66	CLUTS	Color Look Up TableS
67	COO	Chief Operating Officer
68	COPE	Combined Patient Experience, a laboratory medicine database
69	CORBA	Common Object Request Broker Architecture
70	COSI	Corporation for Open Standards International
71	COTS	Connection-Oriented Transport Service
72	COWS	Commission On World Standards, Pathology
73	CPE	Customer Premises Equipment
74	CPEP	Clinical Practice Expert Panel
75	CPR	Computer-based Patient Record: Coronary Pulmonary Resuscitation
76	CPT	Current Procedure Terminology
77	CPU	Central Processing Unit
78	CQI	Continuous Quality Improvement
79	CR	Computed Radiography
80	CSMA/CD	Carrier Sense Multiple Access with Collision Detection, Ethernet features
81	CT	Computed Tomography
82	D-channel	Delta-channel, ISDN channel with 16 kbps bandwidth (see BRI)
83	DDN	Defense Data Network
84	DEC	Digital Equipment Corporation
85	DG	Data General Corporation
86	DHCP	Decentralized Hospital Computer Program, used by DEC and CHCS
87	DHHS	Department of Health and Human Services
88	DICOM	Digital Imaging and Communications in Medicine
89	DINS	Digital Imaging Network Systems - Military term
90	DISA	Data Interchange Standards Association/ASC X12
91	DIX	DEC Intel Xerox, initial standard for Ethernet (now an IEEE 802.3 standard)
92	DMSSC	Defense Medical Systems Support Center
93	DNA	Deoxyribonucleic Acid Sp?
94	DOCking STATION	Doctor Operated Communication Kiosk Intelligently Networking Generalists Synergistically To All The Information Of Need
95	DoD	Department of Defense
96	DoH	Department of Health
97	DRAM	Dynamic Random Access Memory
98	DRG's	Diagnosis Related Groups
99	DSOs	Digital voice channels, used with ISDN
100	DTE	Data Terminal Equipment, usually a computer that interfaces with Ethernet
101	DTS	Dietetics
102	DVA	Department of Veterans Affairs
103	Dx	Diagnosis
104	EDI	Electronic Data Interchange
105	EDIFACT	Electronic Data Interchange for Administration, Commerce, and Transport
106	EGAD	Electronic Grant Application Development project, Dept. of Health and Human Services under NIH
107	ENR	Enterprise Network Roundtable, user group of ATM
108	EOC	Expense Operating Center ? ,an accounting term
109	EOQ	Economic Order Quantity
110	EPI	Enterprise Patient Index
111	EPROM	Electronically Programmable Read Only Memory
112	ESS	Executive Support System
113	FAQ	Frequently Asked Questions
114	FCC	Federal Communications Commission
115	FCS	Full Cover Slip
116	FDA	Federal Drug Administration
117	FDDI	Fiber Distributed Data Interface
118	FIFO	First In - First Out
119	FNA	Fine-Needle Aspiration
120	FOE	Fiber Optic Enclosure
121	FOIRL	Fiber Optic Inter-Repeater Link, used with Ethernet
122	FOMAU	Fiber Optic Medium Attachment Unit, transceiver for Ethernet
123	FTP	File Transfer Protocol
124	FYI	For Your Information
125	GAO	General Accounting Office
126	GATT	General Agreement for Tariff and Trade
127	GHNet	Global Health Net
128	GL	General Ledger
129	GNA	Global Network Academy
130	GNP	Gross National Product

Glossary of Telemedicine and Hospital Information Related Systems Acronyms; 12/10/98

131	Gopher	animated contraction of "go-for" looks for subject or words of interest on the NET
132	GP	General Practitioner
133	GPIP	General Purpose Image Processing
134	GPR	Graphical Patient Record
135	GRIPE	Group for Research in Pathology Education
136	GUI	Graphical User Interface
137	GYN	Gynecological
138	HAF	Hyperalimentation Fluids
139	HCFA	Health Care Financing Administration
140	HCIS	Health Care Information System
141	HCO	Health Care Organization
142	HCTG	Health Care Technology Group
143	HDTV	High Definition Television
144	HEDIS	Health Plan Employer Data and Information Set
145	HIMA	Health Industry Manufacturers Association
146	HIMSS	Healthcare Information and Management Systems Society
147	HIS	Hospital Information System, Health Information System
148	HISPP	Health Informatics Standards Planning Panel, formed by ANSI
149	HISS	Hospital Information Support System
150	HITS	Health Innovations in Technology Systems, yearly award given by the Henry Ford Health System
151	HL7	Health Level 7, Interface Standard
152	HMO	Health Maintenance Organization, Health Maintenance Group
153	HTML	Hyper Text Markup Language
154	IBM	International Business Machines
155	ICD	billing code used for various cases?
156	ICU	Intensive Care Unit
157	ID	Individual Identifier
158	IDN	Integrated Digital Network
159	IEC	Image Exchange Committee, developing Pathology extension to DICOM
160	IEEE	Institute of Electrical and Electronic Engineers
161	IHID	Inter Hospital Image Distribution
162	IIM	Institute for Information Management, Robert Morris College
163	ILCP	International Liaison Committee of Presidents, forum of English speaking pathologists
164	ILO	International Labor Organization
165	ILSG	International Lymphoma Study Group
166	IPA	Independent Physicians Association, or Independent Practice Association
167	ISA	International Standards Association, Instrumentation Society of America
168	ISAM	Indexed Sequential Access Method, (used with data bases)
169	ISDN	Integrated Services Digital Network
170	ISIS	Information System-Imaging System
171	ISO	International Standards Organization
172	IT	Information Technology
173	ITU-T	International Telecommunications Union-Telecommunications, sets ISDN standards
174	JAHIS	Japanese Association of Healthcare Information Systems Industry
175	JPEG	Joint Photographers Expert Group
176	JWG-CDM	Joint Working Group, Common Data Model
177	LAB	Laboratory
178	LAM	Lymphangioleiomyomatosis
179	LAN	Local Area Network
180	LANL	Los Alamos National Laboratory(ies)
181	LEOS	Low Earth Orbit Satellite
182	LIFO	Last In - First Out
183	LIS	Laboratory Information System
184	LM	Laboratory Module, part of an HIS
185	LOINC	Laboratory Observation Identifier Names and Codes
186	LOS	Length Of Stay
187	LSDA	Line Scan Diode Array, provides high resolution large image scanning capability
188	MAC	Medium Access Control, provides access when available from each Ethernet station
189	MAR	Medication Administration Record
190	MAU	Medium Attachment Unit, Transceiver for Ethernet that interfaces between computer and the medium.
191	MC	Medullary Carcinoma
192	MD	Medical Doctor
193	MDC	MUMPS Development Committee
194	MDF	MD Forms
195	MDI	Medium Dependent Interface, Ethernet hardware that connects directly interfaces to the medium.
196	MDIS	Medical Diagnostic Imaging Support system - used by Military.

197	MDL	MD Lookup
198	MDR	MD Retrieval
199	MEDIX	MEDical Data InterX. (X stands for exchange)
200	MGH	Massachusetts General Hospital
201	MHDI	Minnesota Health Data Institute
202	MICHI	Minnesota Institute for Community Health Information
203	MIS	Medical Information Systems, Management Information System
204	MPI	Master Patient Index
205	MRI	Magnetic Resonance Imaging
206	MRI	Medical Records Institute
207	MRN	Medical Record Number
208	MSDS	Message Standards Developers Subcommittee, health care message interchange stds, formed by HISPP
209	MTF	Medical Treatment Facilities - Military Term
210	MUMPS	Massachusetts (Gen. Hosp.) Utility Multi Programming System, Prog. Lang. used by SAIC & some Hosp.
211	NCI	National Cancer Institute
212	NCPDP	National Council of Prescription Drug Pharmacies, National Council of Prescription Drug Programs
213	NCQA	National Committee for Quality Assurance
214	NEMA	National Electrical Manufacturers Association
215	NET	Short for Internet
216	NHS	National Health Services
217	NIGMS	National Institute of General Medical Science
218	NIH	National Institutes of Health, Not Invented Here
219	NIHLB	National Institute of Heart, Lung and Blood
220	NII	National Information Infrastructure, goal to provide equable information services to all Americans
221	NII-HIN	National Information Infrastructure-Health Information Network
222	NINDS	National Institute for Neurological Disease and Stroke
223	NIOSH	National Institute of Occupational Safety and Health
224	NIST	National Institute of Standards and Technology
225	NLM	National Library of Medicine
226	NMF	Network Management Forum
227	NOS	Not Otherwise Specified
228	NPS	Non Printed Specifics
229	NRS	Nursing
230	NSF	National Science Foundation; National Standard Format, for health service claim entries
231	OBRA	the Omnibus Budget Reconciliation Act
232	ODA	Optical Disk Archiving system
233	ODJ	Optical Disk Jukebox, Optical media (platters) for high density digital storage
234	OLE	Object Linking and Embedding
235	OMG	Object Management Group, (responsible for CORBA standards)
236	OOT	Object Oriented Technologies, a Company - does CORBA; Out Of Town
237	OSHA	Occupational Safety and Health Administration
238	OSI	Open System Interconnection, seven layers of hierarchy
239	OT&E	Operational Test and Evaluation
240	PACS	Picture Archiving and Communications System - Used by Military
241	PAD	Patient Administration Department?
242	PAHO/WHO	Pan American Health Organization/World Health Organization
243	PALI	Pathologist Accelerated Laboratory Investigation
244	PAS	Patient Appointment Scheduling
245	PBXs	Private Branch Exchanges
246	PC	Personal Computer
247	PCI	Patient Care Inquiry, high speed buss that carries information in PC's and Power Macs
248	PCM	Personal Computer-Microscope, provides workstation features with digital images
249	PET	Positron Emission Tomography
250	PHO	Physician Hospital Organization
251	PHR	Pharmacy
252	PHS	Public Health Service
253	PM	Pagetoid Melanocytosis, upward discontinuous extension of melanocytes into the epidermis.
254	PMED	Portable Medical Entry Device
255	PO	Purchase Order
256	POE	PowerOpen Environment
257	PPO	Preferred Provider Organizations
258	PRI	Primary Rate Interface-ISDN 23 ea., 64 kbps channels + one 64-kbps D-channel
259	PRO	Peer Review Organization
260	PSA	Prostate-Specific Antigen
261	PtCT	Patient Care Technologies, Inc.
262	QA	Quality Assurance

263	QC	Quality Control
264	QTD	Quarter To Date
265	R&D	Research and Development
266	RAD	Radiology
267	RAID	Redundant Array of Inexpensive Disks
268	RBOCs	Regional Bell Operating Companies, the 7 Baby Bells
269	RBRVS	Resource-Based Relative Value Scale
270	REAL	Revised European-American Lymphoma classification
271	RGB	Red Green Blue, a TV full color generating scheme where all color is obtained by addition of R,G,B
272	RIS	Radiology Information System
273	RLA	Reference Laboratory Alliance
274	RM	Reference Model, Radiology Module, part of an HIS
275	RN, R.N.	Registered Nurse
276	RNA	Ribonucleic Acid
277	RTE	Remote Terminal Emulation
278	RUC	Relative value Update Committee, under AMA, reviews work relative values for effectiveness
279	SAIC	Science Applications International Company
280	SCSI	Small Computer Standard Interface, pronounced "scuzzi"
281	SDC	Surgical Day Care
282	SDOs	Standards Developing Organizations
283	SMTP	? , information protocol
284	SNMP	
285	SNOMED	Systematized Nomenclature of Medicine
286	SOW	Statement of Work
287	SQL	Structured Query Language
288	SRDRG's	Severity Refined Diagnosis Related Groups
289	SSN	Social Security Number
290	STARPAHC	Space Technology Applied to Rural Papago Advanced Health Care
291	T1	Communication lines with 1.54Mbt/sec transmission rate
292	TA	Terminal Adapter, interfaces with ISDN
293	TAMC	Tripler Army Medical Center
294	TCP/IP	Transfer Control Protocol, Internet Protocol
295	TDS	? , Total Dissolved Solids
296	TE	Terminal Equipment, devices using ISDN to transfer information
297	TELNET	Information Protocol
298	TIFF	Tagged Image File Format, Popular image file format for multiple platforms.
299	TM	Telemedicine
300	TQM	Total Quality Management
301	TRP	Technology Reinvestment Project
302	UHC	University Hospital Consortium
303	UR	Utilization Review
304	URL	Universal Resource Locator
305	USDHHS	United States Department of Health and Human Services - HCFA
306	VA	Veterans Administration
307	VAR	Value Added Reseller
308	VRAM	Video Random Access Memory
309	WAN	Wide Area Network
310	WASP	World Association of Societies of Pathology, Anatomical and Clinical, White Anglo-Saxon Protestant
311	WCP	World Congress of Pathology
312	WHIN	Wisconsin Health Information Network
313	WHO	World Health Organization
314	WOM	Write Only Memory, Useful for storing your mother-in-law's address
315	WORM	Write Once Read Many - Type of memory
316	WSU	Work Storage Unit, usually very high density digital storage may have fiber optic data transmission.
317	WTO	World Trade Organization
318	WWW	World Wide Web, graphical interface with hypertext used on the NET
319	XIWT	Cross-Industry Working Team, working on framework for the National Information Infrastructure
320	YTD	Year To Date

APPENDIX C

1997 ANNUAL REPORT
FOR
DAMD17-94-J-4500

TABLE OF CONTENTS

APPENDICES

1. INTRODUCTION

This is the 1997 Fiscal Year Annual Report for grant DAMD17-94-J-4500 (Dual-Use Telemedicine Support System for Pathology) from the USAMRMC (U.S. Army Medical Research and Materiel Command) of Ft. Detrick, Maryland. The research reported here involves upgrading the technology of the two telepathology workstations (Figure 1) previously built under this grant and the design and fabrication of the more compact PC Microscope. This research was conducted in parallel with NIH (National Institutes of Health) grant 5 R44 GM44420-03 (Image Handling System for Pathology and Telepathology) and contract DAAH01-95-C-R209 (Workstation for Medical Images) issued by the U.S. Army Missile Command (Redstone Arsenal, Alabama) and sponsored by DARPA (Defense Advanced Research Projects Agency).

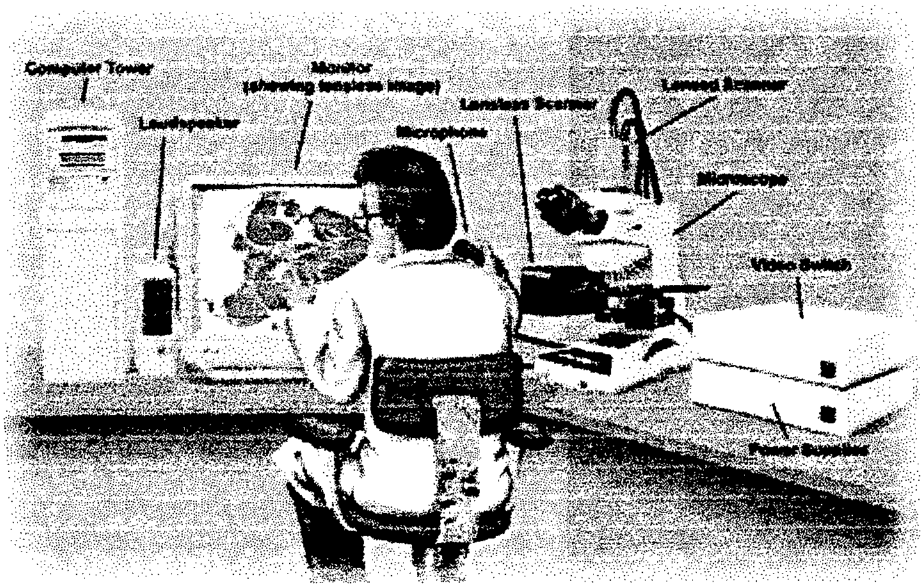


Fig. 1 - Kensal's Telepathology Support System (TSS).

The sections in this report are: (1) **Introduction**. (2) **TSS (Telepathology Support System) Retrofit** describes technology upgrades currently being made to the two existing Ft. Detrick workstations. (3) **PCM (Personal Computer Microscope)** describes the progress being made on the more compact, Macintosh telepathology workstation being built using Kensal's lensless/lensed technology. (4) **Field Trials** gives information on the first telepathology experiment using the Ft. Detrick workstations and plans for a future field trial. (5) **Virtual Microscopy** describes Kensal's self-running multimedia CD-ROM database of telepathology cases that were created using the Lensless/Lensed Imaging technique. (6) **Summary** includes positive and negative aspects of the project over the last year, recommendations for extending our project, and a conclusion.

2. TSS (TELEPATHOLOGY SUPPORT SYSTEM) RETROFIT

The TSS consists of standard, off-the-shelf components. A user interface permits production of a lensless "scout" image of the entire coverslip of a glass microscope slide. Using the scout image as a reference, areas of interest where finer detail is needed to complete the diagnosis can be magnified using traditional lensed microscopy.

This breakthrough is due to a patented development filed by Kendall Preston, Jr. (former President of Kensal Corporation) in the early 1980s wherein lenses are not required to generate a low-resolution magnified image. Instead, a line scan diode array (LSDA) is employed with the finest possible spacing between light detectors. Light pushing through the tissue sample produces a shadow detected by the LSDA. The precision of this shadow image depends only on the spacing of the diodes in the diode array and of course to their sensitivity to the impinging light and to the scan rate of the LSDA itself.

In late 1996, a dramatic improvement in low-resolution lensless microscopy was made possible by the introduction of the new EG&G Reticon RL4000P and the Kodak KLI-10203CA diode arrays which have diodes spaced on seven micrometer centers (Figure 2). Previous offerings from both manufacturers have diodes spaced on thirteen micrometer centers (Figure 3). The seven micrometer diode array makes it possible to digitize full coverslip images at 20 thousand picture points per square millimeter. With the thirteen micrometer diode array, only 6 thousand picture points per square millimeter was possible.

Currently, the two Ft. Detrick workstations are being retrofitted to incorporate the Kodak diode array with a seven micrometer spacing. The changes which are being made to upgrade the workstations to accommodate the new diode array include: Replacing the image acquisition card in the computer with the recently introduced Genesis card from Matrox. The Genesis card utilizes the PCI (Peripheral Component Interconnect) bus and has the capability of handling the higher data transfer rates imposed by the new diode array. The operating system of the computer has been upgraded to Windows NT v.4.0 to accommodate the Genesis card and improve system operation. The major modifications are to the lensless scanner and include replacing the Kodak KLI-4103 diode array with the KLI-10203 diode array and replacement of the Kodak KLI-4103EB evaluation electronics board with a custom electronics board (being designed and fabricated by Kline Research of Reseda, California). This custom electronics board has significant performance improvement over the Kodak unit. Some mechanical redesign of the scanner assembly is required to accommodate the KLI-10203 and the new electronics board.

Enhancements have been made to the system software to simplify the user interface and to make the workstation more "user friendly". These enhancements are a direct result of the 1996 Luke/Mayo Telepathology Study. (See Section 4.) Current capabilities of the TSS are explained fully in Appendix A, TSS User Manual.

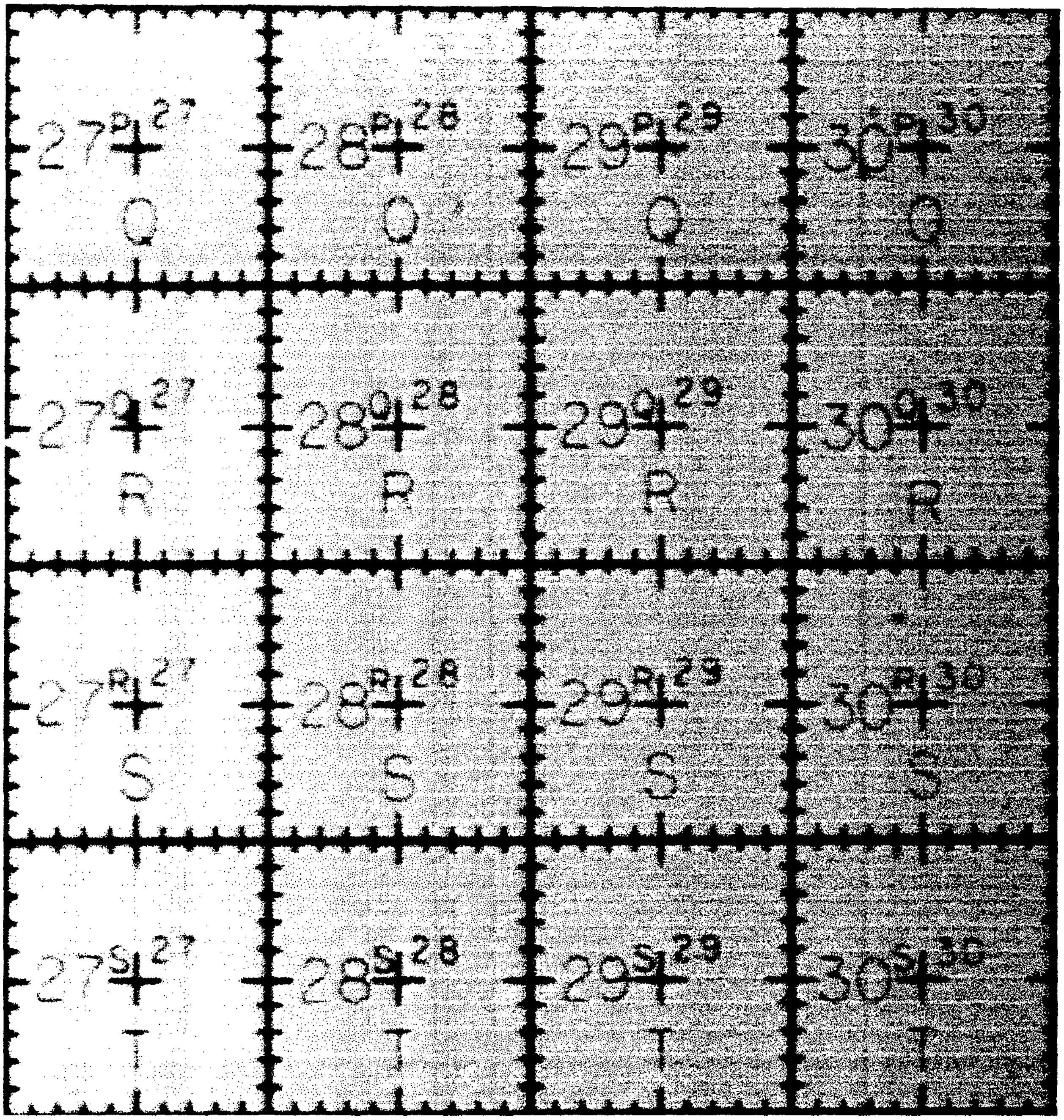


Fig. 2 - Seven micrometer lensless scan of a portion of a Lovins field finder.

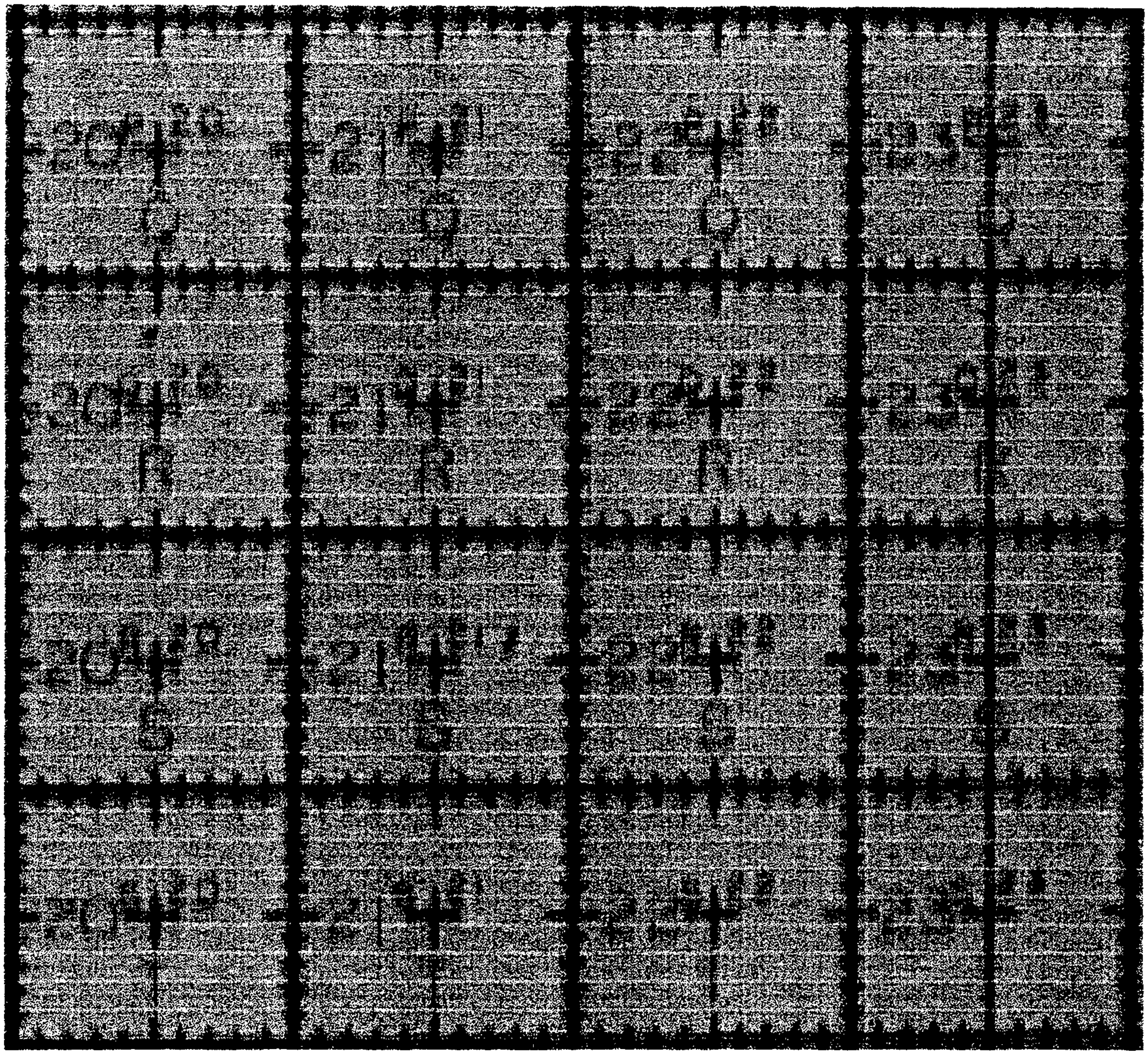


Fig. 3 - Thirteen micrometer lensless scan of a portion of a Lovins field finder.

3. PCM (PERSONAL COMPUTER MICROSCOPE)

The Windows NT-based TSS produced for the U.S. Army is composed of standard, off-the-shelf components. This system occupies an entire desktop. The more compact Macintosh-based PCM being produced for the U.S. Army has been reduced in size. See Figure 4. This extraordinarily simple and compact mechanism will provide a PC (Personal Computer), the lensless microscope, and a lensed microscope in a single housing.

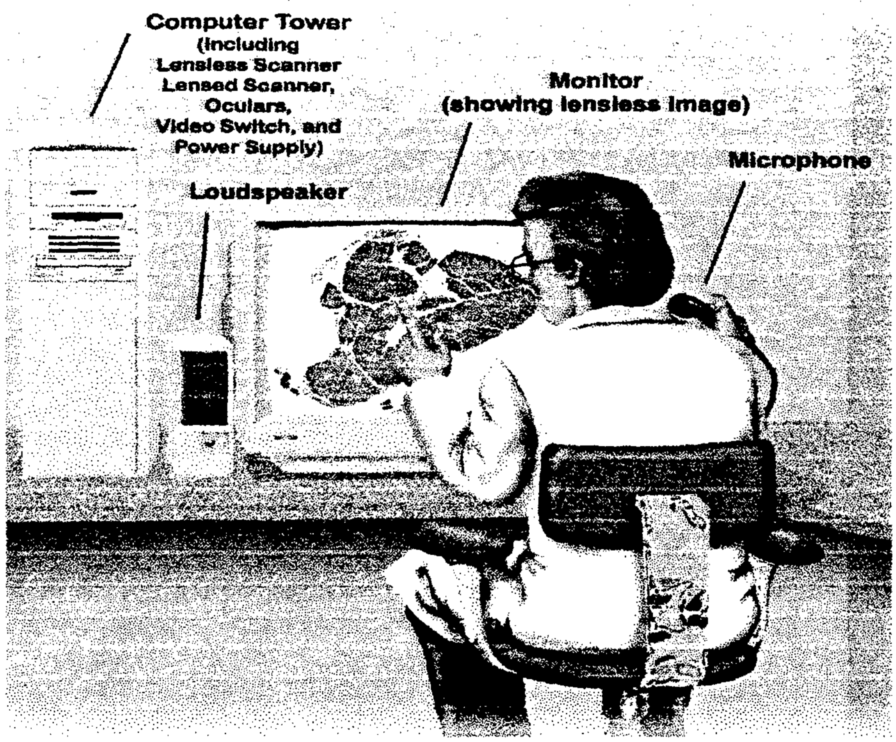


Fig. 4 - Kensal's Personal Computer Microscope (PCM).

In use, the PCM will receive the microscope slide through an insertion guide into a fixed, monolithic slide holder. A single traction roller will grab the slide, move it past the lensless scanner and seat it in the slide holder where it will be secured by both the traction roller and an associated pinch roller. Two moveable members, both of which have cutouts to accept the immobile, monolithic slide holder, would be used to position high-resolution, continuously-focusable optics if required. These members would consist of a horizontal yoke which would have ±1" travel so as to examine the full extent of the coverslip and the vertical platform with a ±1/2" travel to cover the 1" vertical dimension of the coverslip. Light will be delivered through a light-pipe illuminating a circular area approximately 2.5mm (0.1") in diameter and a lensed CCD scanner used if imaging at submicron resolution is required.

The PCI Stepper Motor Control PCB (Printed Circuit Board) for the Macintosh-based PCM has been designed and is being fabricated. This work included schematic capture, Altera HLD (High Level Design) entry and simulation, PCB layout, fabrication, assembly and design verification. We are proceeding with system integration of the Frame Capture Board (FCB), the Camera Head Electronics (CHE), and the Motor Control Board (MCB) into the PCM.

Electronic imaging hardware has been designed and developed to be used in the PCM. Two test enclosures for the PCM camera have been designed, developed and fabricated.

Fabrication drawings for and initial testing of the optical and mechanical systems have been completed. We are proceeding with construction of the optical and mechanical systems.

The logic and interface necessary for each of the user modes have been implemented (See Appendix B.). For each of the modes, the preferences file was required and the dialog and interface needed for the preferences was developed. For the transcription modes the voice dialog was made re-sizable, and the database entry dialog was executed.

4. FIELD TRIAL

In early 1996 a telepathology experiment began involving the Kensal Corporation (Tucson, AZ), the Mayo Clinic (Scottsdale, AZ) and the 56th Medical Group at Luke Air Force Base (Litchfield Park, AZ). The pathologists who participated in this experiment were Louis H. Weiland, MD and Kevin Leslie, MD from the Mayo Clinic, and Hermilando Payen, MD and Felix Mamani, MD from Luke AFB. A considerable amount of time was spent during the field trial dealing with hardware, software, and ISDN problems which became apparent from continued use of the equipment.

The following information is a result of the Luke/Mayo 1996 Telepathology Study.

A total of 42 cases were completed - 27 from Mayo and 15 from Luke. Five organ systems were represented in the field trial: Immune System, Breast, Skin, Excretory System, and Female Reproductive System. In 93 percent of the cases, diagnoses arrived at by using Kensal's TSS were the same as or similar to the original diagnoses arrived at by using traditional microscopy.

From the statistics provided to Kensal by our expert pathologists (See Appendix A.), on the average, 16 AOIs (Areas of Interest) were selected from each scout image and the diagnosis was determined by looking at 15 of those AOIs. The majority of diagnoses were arrived at while viewing a 40x or 20x high-magnification image.

Conclusions drawn from the Luke/Mayo 1996 Telepathology Study are as follows: (1) The TSS does permit full-specimen imaging that is now totally impossible using the traditional microscope. (2) Full-specimen digital images may be displayed on the computer screen in full color. (3) These digital images may be successfully transmitted for remote consultation by ISDN (Integrated Systems Digital Network). (4) Digitized pathology cases may be conveniently archived for future reference.

5. VIRTUAL MICROSCOPY

As part of our work on both PCM and TSS, a large image library has been produced. Our staff has modified the software package called "Virtual Microscopy".

Virtual Microscopy is a self running multimedia CD-ROM database containing telepathology cases, created using the L/L (Lensless/Lensed) Imaging technique. Each case includes a Lensless, low resolution scout image, several regions of interest recorded by lensed microscopy (referred to as High Magnification image or "HM"), and recorded-voice diagnosis.

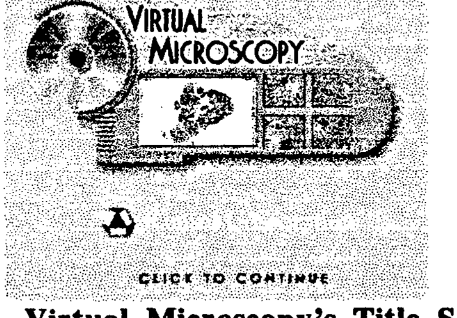


Fig. 5 - Virtual Microscopy's Title Screen.

5.1 Mannequin Search

Pathology cases are accessed interactively by selecting call-outs on a mannequin. Rotating or changing the gender of the mannequin allows the user to access all pathology cases.

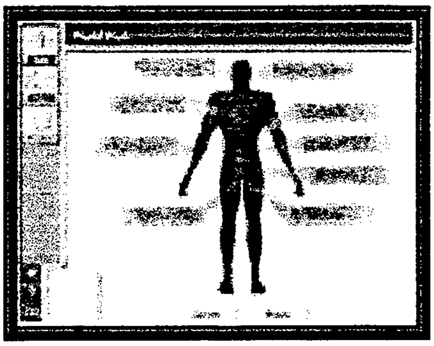


Fig. 6 - Virtual Microscopy's Mannequin Search screen.

5.2 Index Search

Cases are listed alphabetically by tissue or organ system in "Index" search mode. Selecting a case displays thumbnails of all images and voice playback options for that case. Clicking on a thumbnail image displays the selected scout or HM at full resolution.

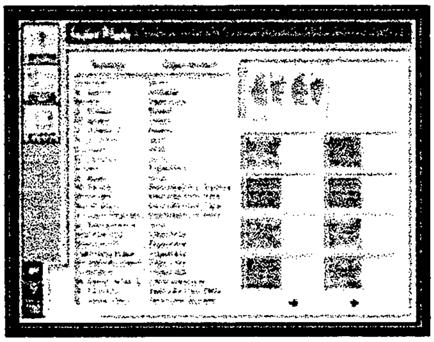


Fig. 7 - Virtual Microscopy's Index Mode screen.

5.3 Scout View

When a case has been selected with the interactive mannequin or with the indexed listing, the case scout image is displayed. In this display, selected regions of interest are outlined in black. Clicking on a region of interest opens the corresponding HM. Pressing the playback button in this display starts the voice-over for the scout image.

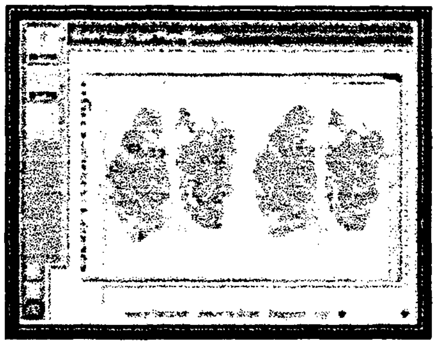


Fig. 8 - Virtual Microscopy's Scout Image Low Magnification screen.

5.4 Lensed High Magnification

Selecting a region of interest on the scout image, or selecting the HM thumbnail in the indexed listing displays the corresponding lensed, HM image. The HM image can be expanded to full resolution, panned, or the HM's voice-over can be played back.

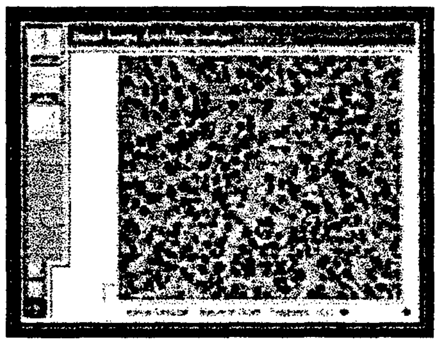


Fig. 9 - Virtual Microscopy's Lensed Image High Magnification screen.

5.5 Slide Show

The 'Slide Show' screen lists the images in the current carousel and allows the user to view, edit, or play the carousel contents.

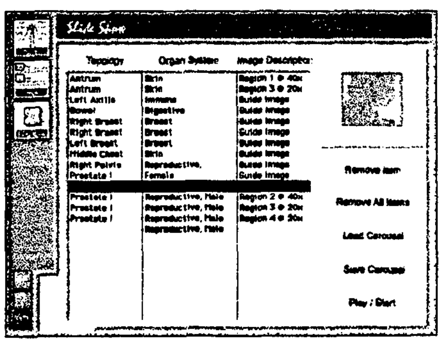


Fig. 10 - Virtual Microscopy's Slide Show assembly screen.

The 'Playback' screen allows the user to navigate through the selected images, zoom and pan on an image, or play the voice diagnosis. At any time the user can stop the playback and return to the 'Slide Show' screen.

Fig. 11 - Virtual Microscopy's Slide Show playback screen.

"Virtual Microscopy" will be an effective educational tool in that it simulates the operation of using lensless and lensed images. It was designed in a way that course material may be distributed inexpensively by CD-ROM or over the Internet.

6. SUMMARY

This section includes the positive and negative aspects of this project over the last year, a request for extending this project, and a conclusion.

6.1 Positive Aspects of Grant Research in FY 96-97

The rapid prototype workstations (TSS) proved that lensless imaging is an effective tool for the pathologist. As a result of this telepathology study, "Virtual Microscopy" was created (Section 5) and the need for software enhancements was realized and implemented (Section 2).

The TSS User Manual has been completed and updated to include all retrofits to date. An installation guide is now being written by Kensal staff.

Work is proceeding on schedule for the PCM. Delivery date of the finished product is expected by the end of December 1997.

In June 1997, Kensal attended the DARPA Workshop held in Tucson, Arizona. Kensal's presentation was received with much excitement and enthusiasm. Dr. Richard Satava was especially impressed with Kensal's "Virtual Microscopy" and the educational possibilities it holds.

In August 1997, Kensal staff was invited to Ft. Detrick, Maryland to discuss our telepathology project. General Zajtchuk expressed interest in having Telemedicine Research Laboratory involved in the next telepathology experiment using Kensal's workstations.

In September 1997, Kensal's President and CEO, Diane Conti, visited AFIP (Armed Forces Institute of Pathology) and Harvard Medical School. Both AFIP and Harvard Medical School were extremely enthusiastic about our project and expressed interest in being chosen for future telepathology experiments.

6.2 Negative Aspects of Grant Research in FY 96-97

Due to problems with hardware, software and ISDN hook-up, the Luke/Mayo 1996 Telepathology Study was unable to achieve all that was initially planned. Time did not allow for the actual glass microscope slide to be reviewed by the remote user so we referred back to the patient file for the original diagnosis of each slide which was done by traditional microscopy. However, the experiment did prove successful in that we were able to establish the fact that the TSS is a viable diagnostic tool for the pathologist.

Problems with Matrox shipping new products has caused a delay in retrofitting the TSS workstation since the second quarter of 1997. The current ship date is the end of October 1997. We have on loan from Matrox a demonstration board that will allow us to proceed with the retrofit until the boards on order become available.

6.3 Extending Our Research

Under grant DAMD17-94-J-4500 from the U.S. Army Medical Research and Materiel Command (USAMRMC, Ft. Detrick, Maryland) and contract DAAH01-95-C-R209 from the Redstone Arsenal (Redstone, Alabama), the Kensal Corporation has built three lensless/lensed TSS workstations - two for Ft. Detrick and one for Redstone Arsenal and designed and initiated fabrication of two PCMs.

In September 1997, Kensal Corporation received permission from Redstone Arsenal to extend the completion date of that contract to June 30, 1998. This will allow us to retrofit the one TSS workstation that has been built under DAAH01-95-C-R209 for Redstone Arsenal into a revolutionary new instrument that will combine recent advancements in diode array scanning with the existing instrument that was completed during late 1996 and will make it compatible with the two Ft. Detrick workstations now undergoing the same retrofit. It will also allow sufficient time to complete and test the PCM.

As this contract works in conjunction with the Detrick grant, we request an extension of time for our USAMRMC grant to June 30, 1998. This time extension is due in large part because of delays in delivery of components (see Section 6.2). This would allow us to complete all retrofits, complete the two PCMs, and establish a solid experiment using all three retrofitted TSS workstations and the two PCMs. No additional funds are required.

6.4 CONCLUSION

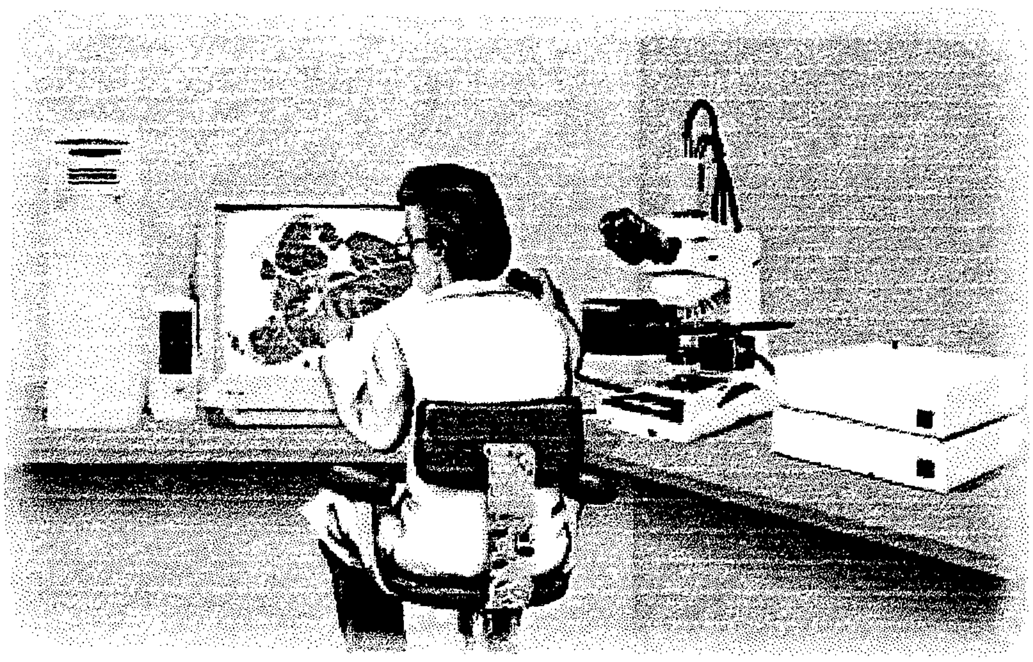
Proving that lensless microscopy <u>does</u> work with the TSS, progress made on the PCM, and creating "Virtual Microscopy" have been exciting developments for FY 96-97. Testing the retrofitted TSS, successfully completing our first transmission on PCM, and fully developing "Virtual Microscopy" are a few of the things we are looking forward to in FY 97-98.

But, as the old saying goes, "with the good comes the bad". That has certainly been the case at Kensal this year. We were all deeply saddened by the death of Kendall Preston, Jr., President of Kensal Corporation, and inventor of lensless microscopy. Although this loss has been great, Kensal staff is all the more dedicated to seeing this project through to its successful completion. Because of his expert planning and leadership, Kensal staff can confidently complete this project's final phase and continue to develop lensless microscopy into a viable replacement for traditional microscopy.

APPENDIX A

TSS USER MANUAL

MULTI-USE
TELEMEDICINE SUPPORT SYSTEM
FOR PATHOLOGY
(TSS)



USER MANUAL

Kensal Corporation
Tucson, AZ

USER MANUAL
FOR THE
MULTI-USE TELEMEDICINE SUPPORT SYSTEM FOR PATHOLOGY
(TSS)

TABLE OF CONTENTS

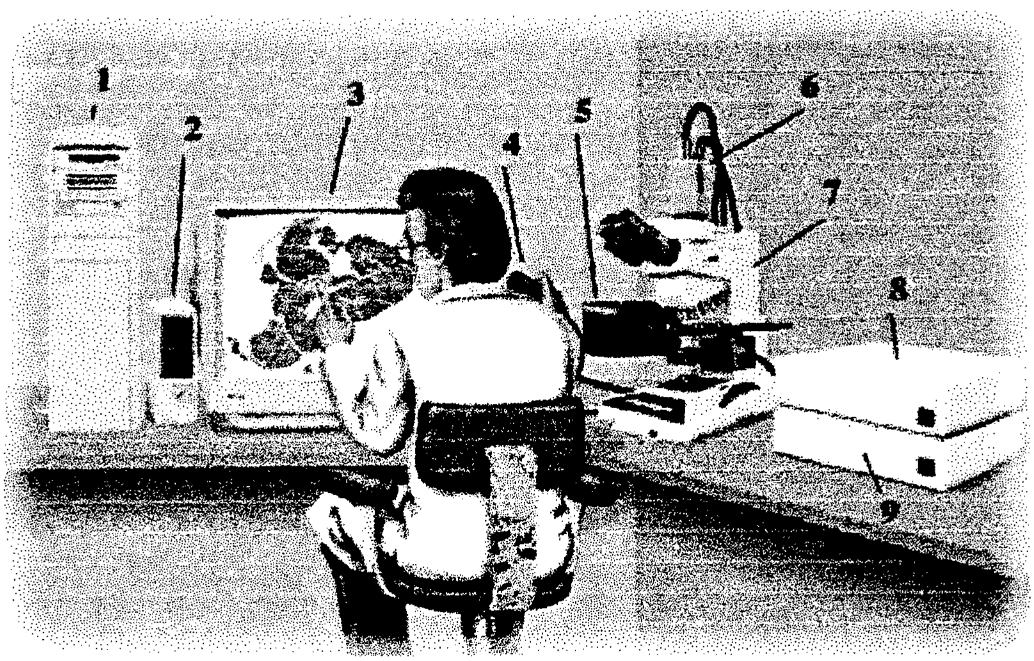
USER MANUAL
FOR THE
MULTI-USE TELEMEDICINE SUPPORT SYSTEM FOR PATHOLOGY
(TSS)

1. INTRODUCTION

Welcome to the Multi-Use Telemedicine Support System for Pathology (TSS). The TSS is a medical imaging telecommunications system that aids the pathologist in viewing glass microscope slides during routine pathological examinations, and allows for sending and receiving images and voice files.

The TSS User Manual is designed to serve as a reference tool for users who are unfamiliar with the standard operations of the system. It provides the user with step-by-step instructions on how to operate this workstation. This is a user manual, NOT a technical manual. For technical information, please contact Kensal Corporation.

1.1 Workstation Components



1 - Computer Tower
2 - Loudspeaker
3 - Touchscreen Monitor
4 - Microphone
5 - Lensless Scanner
6 - CCD Camera
7 - Microscope
8 - Power Supplies
9 - Motion Controller

2. GETTING STARTED

The TSS consists of identical workstations that capture, display, store, retrieve and communicate pathology images over ISDN (Integrated Services Digital Network) lines. For ease of use, all modes of operation are accessible by simply touching the appropriate buttons on the touchscreen with your finger-tip. A white arrow appears on the screen indicating where you have touched last. If you drag your finger across the screen, the arrow will follow your movement. Releasing pressure on the screen will activate the button underneath the arrow.

This instrument is designed for use on glass microscope slides with coverslips <u>only</u>. Scanning any wet or non-solid medium can result in serious damage to the workstation.

2.1 Turning On Workstation

To start the workstation, push the switch on the CPU to the "On" position. After a few seconds, the loader screen with the operating system will appear.

2.2 Choosing an Operating System

Touch ***Windows NT Workstation***and press "Enter" on the computer keyboard to choose the operating system. This is the first default option and will automatically connect after 30 seconds. Wait for the *Windows NT Welcome* Screen to appear.

2.3 Logging On

When prompted, press "Ctrl + Alt + Del" simultaneously on the computer keyboard to log on to the system. A *Welcome* dialogue box will appear asking for the username and the password. (NOTE: A Kensal technician will set up your username and password before you begin using this system.) After typing in your username and password, touch ***OK***. After a few seconds the *Main TSS Window* will appear. **The TSS is now ready to conduct examinations in either the Local or the Remote Mode.**

3. LOCAL MODE

Local Mode is used when the operator is 1) making an examination of a glass microscope slide for use at his/her location, 2) making a scout image to transmit to a remote workstation, and 3) recalling stored images for examination or comparison.

The operator will have the option of ending the session at any time by selecting ***Home***, which initiates a return to the *Main TSS Window*. From the main window, press ***Exit*** to prepare the system for shut down.

The operator may press ***Help*** to access the on-line Help Index. By touching any of the green, underlined commands, the corresponding help file will appear on screen. To return to the help index, touch ***Back***, found on the gray menu bar. Browse the help application by pressing on the ***Search*** option. To exit Help, touch ***File*** on the white menu bar, and when the menu drops down, touch ***Exit*** to return to the *Main TSS Window*.

3.1 Initiating Local Mode

Touch ***Single Station New Image Send and View***. This will initiate the Local Mode.

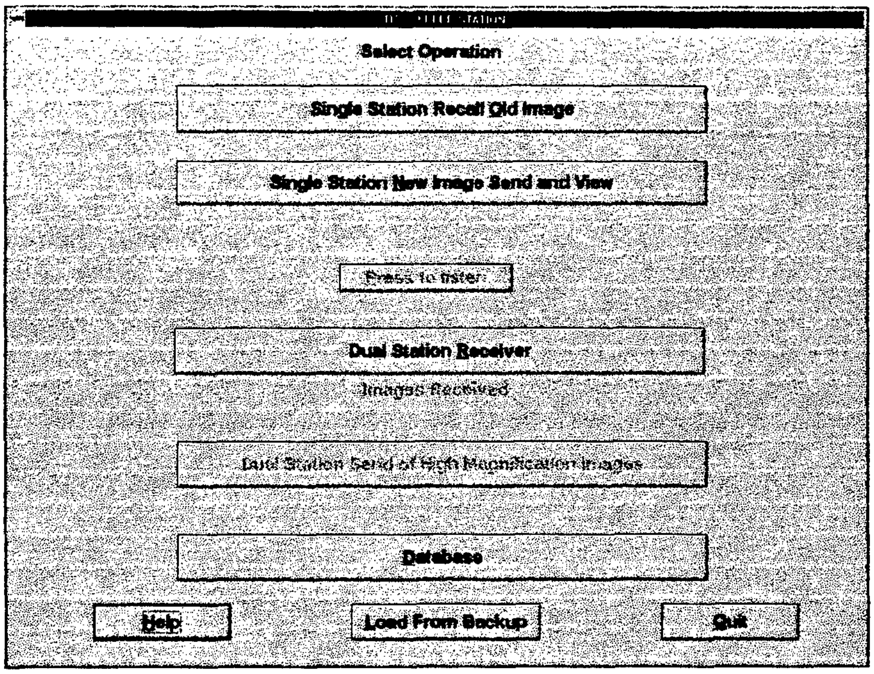


Fig. 2 - The Main TSS window.

3.2 Entering Slide Number

Enter the six digit accession number of the microscope slide by using the numerical keypad located on the touchscreen. If a mistake is made, touch ***Clear*** and reenter the numbers. If the number was entered correctly, touch ***O K***. The *Load Slide Carrier* window will appear.

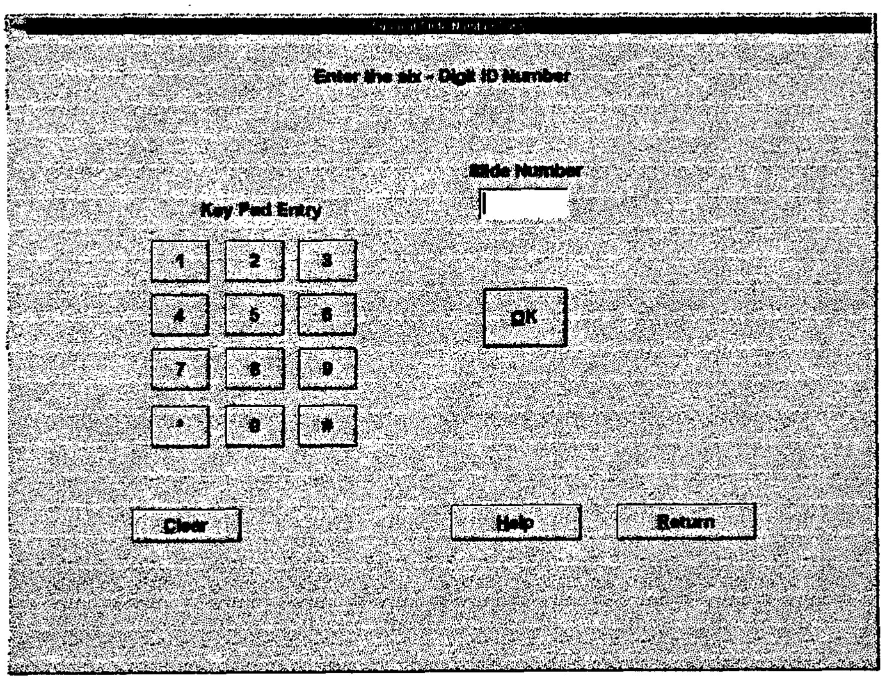


Fig. 3 - The Surgical Slide Number Entry screen.

3.3 Loading a Microscope Slide

Clean the microscope slide of all fingerprints and markings as these may affect the appearance of the scanned image. Place the microscope slide, cover glass facing up, with the labeled end toward the back of the microscope. Guide the glass slide into position and secure it in place with the specimen holder, as shown in Figure 4 below.

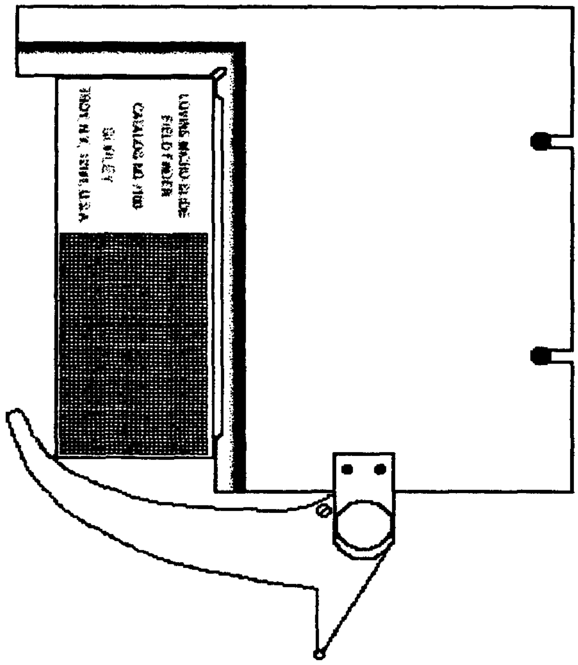


Fig. 4 - Drawing depicting correct slide orientation and security by the specimen holder.

Once the slide is securely in place, choose which stain has been applied to the tissue. The correct slide number should appear in the upper right corner of the screen. Touch ***O K***. The microscope will initiate the scanning to produce a scout image of the slide.

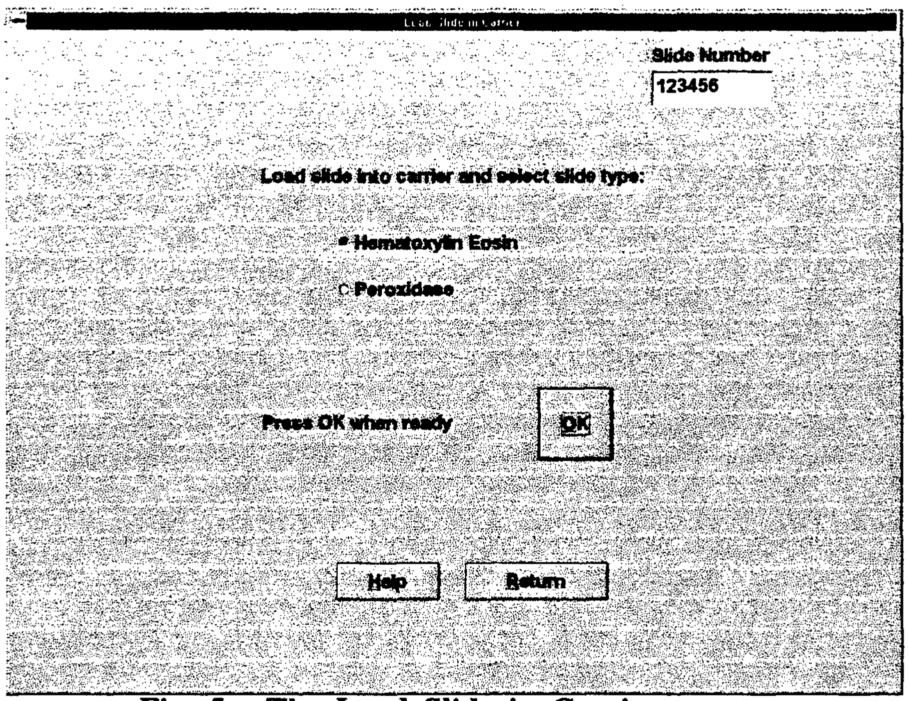


Fig. 5 - The Load Slide in Carrier screen.

3.4 Viewing the Scout Image

The *Scout Image Display, Select HI-MAG* screen will appear displaying the scout image in the main window. The scout image is automatically saved when the image is loaded and focused automatically. There is no way to further adjust the focus of a scout image. A thumbnail will appear in the upper right corner of the screen showing the current location on the scout image in a green box. The operator has the ability to pan and scroll around the image through "jumping," a process which will move any touched region of the image to the center of the screen.

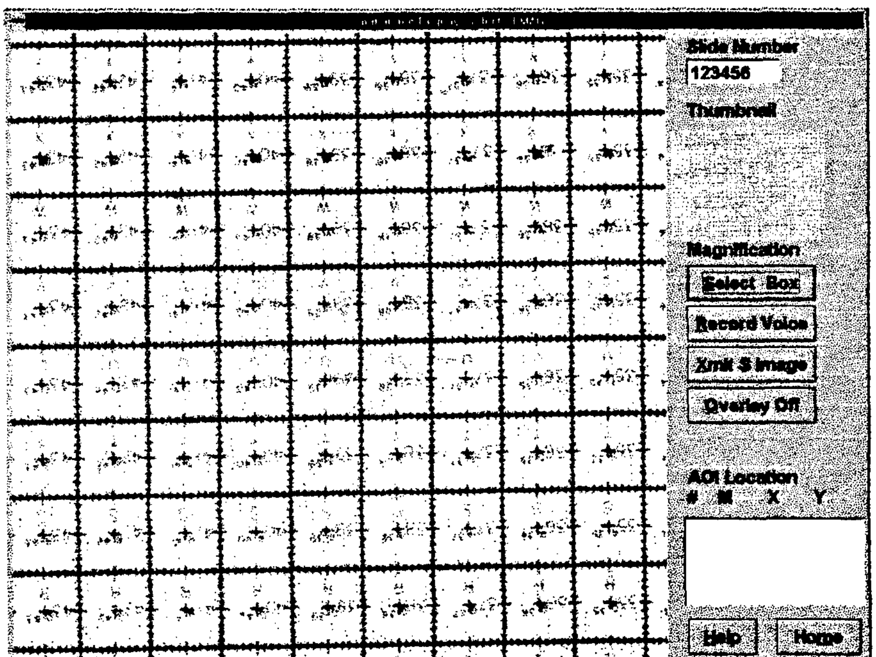


Fig. 6 - Example of the Scout Image Display, Select HI-MAG screen.

The user may record any comments about the scout image by pressing ***Record Voice.*** A dialogue box will appear, making sure that the user is ready to record. Upon pressing ***Yes,*** a new dialogue box will appear, immediately initiating the recording process. When finished recording the message, touch ***Stop***. To replace the message, touch ***Restart*** and begin again. When finished, touch ***Stop*** and then touch ***Exit.***

3.5 Selecting Locations for Higher Magnification

While viewing the scout image, the operator can identify areas of interest (AOI) for which a high magnification image is desired. To select locations for high-magnification examination, touch ***Select Box***. This will bring up a dialogue box containing buttons for magnifications ranging from 2x to 40x.

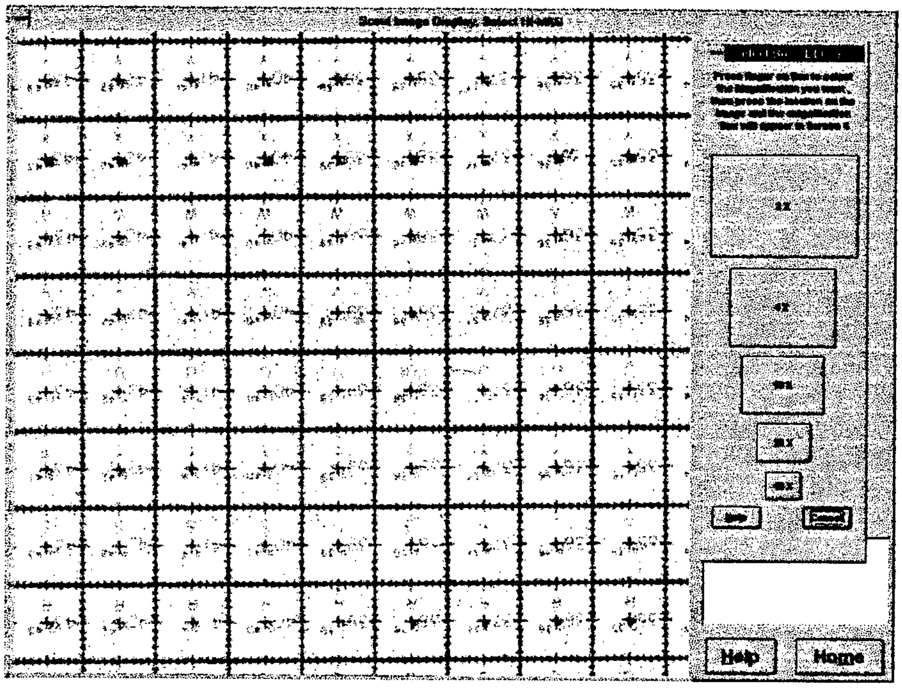


Fig. 7 - Example of the dialogue box used to highlight AOI's and their subsequent magnifications.

Touch the magnification desired for an area of interest. A small, square, black marker will appear on the screen in place of the white arrow marker. Touch the location on the scout image to be magnified. A black box surrounding the area to be magnified will appear. If the box needs to be moved to a more specific location, use the arrow keys on the keyboard to move left, right, up, or down. **Once the screen is touched again, the AOI box will turn green and become fixed in that location.** The size of the box surrounding the AOI will correlate with the desired magnification: a large box represents a low magnification while a small box represents a high magnification. In addition, the coordinates and the chosen magnification of the area will appear in the AOI Location Box in the lower right corner of the screen.

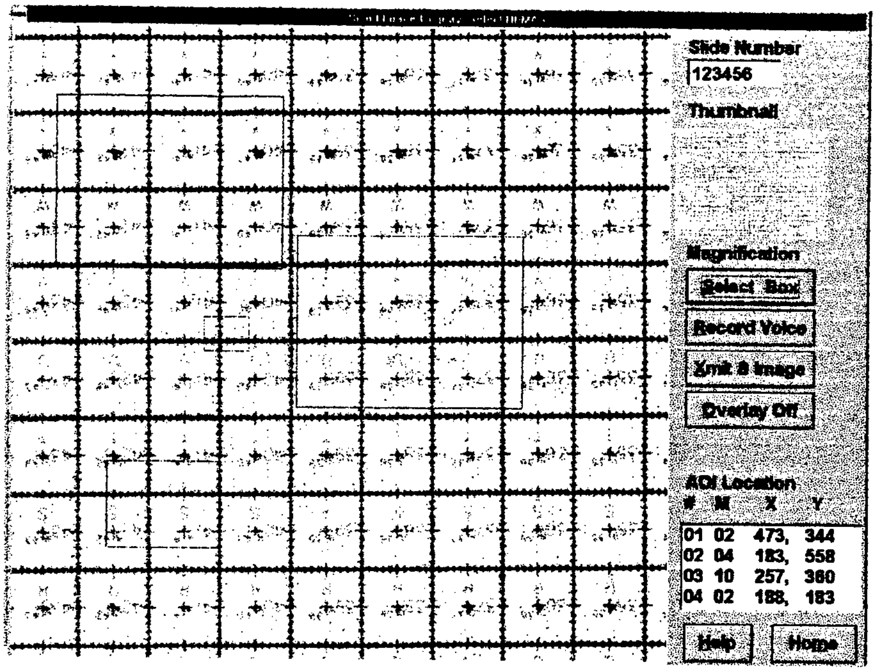


Fig. 8 - Example of a scout image with selected regions for higher magnification.

The operator may continue to select locations and magnifications as desired. Touching the ***Overlay*** button will clear the scout image of green boxes for ease of viewing. Touching ***Overlay*** again will bring the green boxes back on the screen.

3.6 Viewing and Saving High Magnification Images

When all desired locations have been selected, touch inside of any green box to retrieve the corresponding high magnification image. The system will switch from the lensless scanner to the lensed microscope. The computer will move the slide to the desired X,Y location, rotate the objective turret to the proper magnification for the location selected, adjust the condenser for the corresponding objective, adjust the illumination level, and display the high-magnification image on the monitor in real time. Unlike the lensless scans, the microscope showing the higher magnification images does not automatically adjust the focus.

The *High Magnification Image Window* will appear with the first selected high magnification image in the large window. A thumbnail of the scout image will appear in the upper right corner showing, with cross-hairs, the current location on the slide. The operator may use the touchscreen display to change objectives, adjust the focus, and examine the slide under the microscope using the "jump" method mentioned earlier. If the light needs additional adjusting, it may be helpful to turn the apperture at the center, base of the microscope. Open or close the apperture to let in the appropriate amount of light.

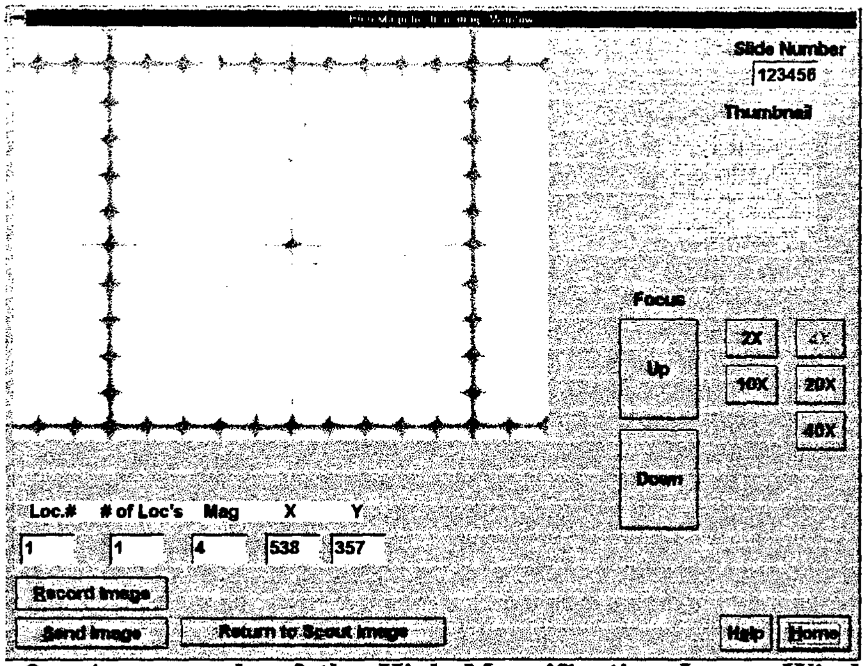


Fig. 9 - An example of the High Magnification Image Window.

Once an AOI, magnification, and focus have been set, the operator may save the image by touching ***Record Image***. The computer will compress the image using a JPEG format and store the image with a file name of the microscope slide identification number and an extension in numerical order. The computer will also store the date, time, and location of capture.

The option to record a verbal message will appear when saving a high-magnification image. To record a message, touch ***Yes***. Recording will start automatically with the elapsed time appearing. When finished recording the message, touch ***Stop***. To replace the message, touch ***Restart*** and begin again. When finished, touch ***Stop*** and then touch ***Exit***.

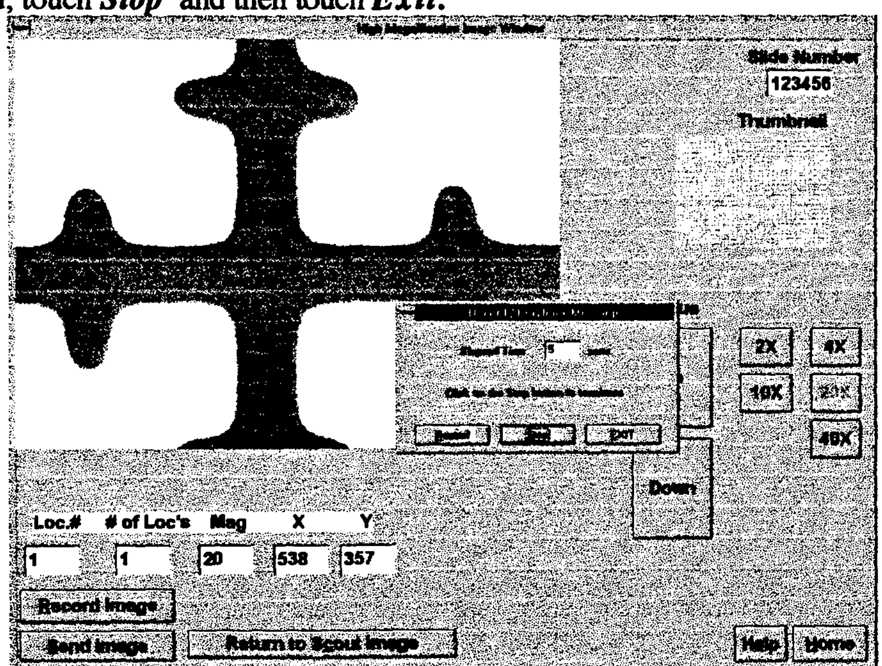


Fig. 10 - An example of the Record/Playback Message box used to record and playback comments about the image being displayed.

To move to the next X,Y location, the user must first return to the scout image by touching ***Return to Scout.*** To select an area of interest, simply touch anywhere inside the green box which frames the desired area and the corresponding high magnification image will appear. Repeat steps in Section 3.6 to view and save any of the high magnification images.

If you have placed one green box inside of another, you can select the larger of the two boxes by touching inside of its borders but outside the border of the smaller box. Choose the small box by touching as close to its center as possible.

The operator has the option to return to the scout image to select additional areas for high-magnification images by touching ***Return to Scout*** *Image.* The system will return to the *Scout Image Display, Select HI-MAG* screen, where the operator can select more areas of interest to be magnified. Don't forget to save any important images by pressing ***Record Image.***

When finished with the examination, touch ***Home*** to return to the *Main TSS Window.*

3.7 Recalling Stored Images

To recall and view stored images, touch ***Single Station, Recall Old Image*** from the main menu. The computer will provide a list of all scout images that have been stored. The operator may select any scout image and its associated high-magnification images to be recalled and viewed.

To view a scout image, select the number of the scout image you wish to view by using the scroll bar to the right of the *Scout Image* window. Touch ***View Scout*** when the desired slide number is highlighted in blue. The date, time and source of capture will be displayed beneath the thumbnail.

Once an image is chosen, there is the option to transmit a scout image from the scout image display screen by pressing the ***X-mit S Image.*** When you return home, the system will request that you chose a receiving location. A progress bar will document the transmission process.

To view a high-magnification image you must return to the *Recall Stored Images* screen and select the number of the high-magnification you wish to view by using the scroll bar to the right of the *Hi-Mag Images* window. Touch ***View High Mag.*** When viewing a stored image, there is no way to jump around the image, or adjust the light or the focus, since the frame showing on screen has been recorded as is. The image may be transmitted by touching ***Send*** Image and following the prompts. By touching ***Back***, the operator will return to the *Recall Stored Images* window.

The only way to access additional saved regions is by returning to the main list of stored images, and selecting the corresponding number directly from the main list.

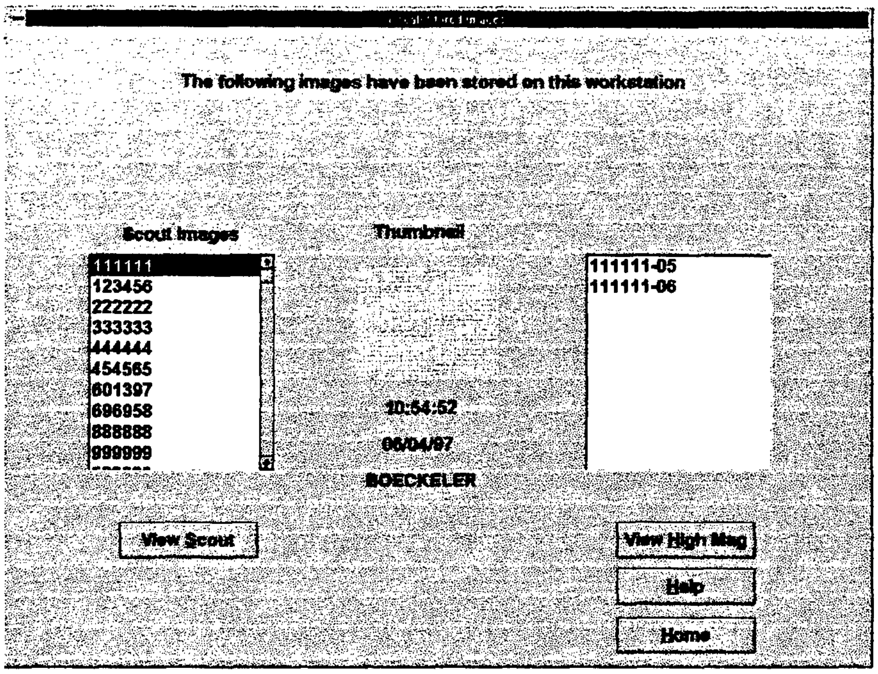


Fig. 11 - An example of the Recall Stored Images screen.

4. REMOTE MODE - SENDING

Remote Mode - Sending is used when the operator wishes to transmit an examination to a remote location for evaluation. The sending station's roles are (1) to create the initial scout image by scanning the microscope slide, and, at the request of the receiving station, (2) make and transmit high-magnification images of locations selected by the receiving station

4.1 Preparing a Scout Image for Transmission

Capturing a scout image for transmission should be done in the same manner as described in the Local Mode section. Touch ***Single Station New Image Send and View.*** Continue as directed in Sections 3.1 - 3.4, by entering the slide accession number, loading the microscope slide, and waiting as the scanner produces a scout image of the slide.

4.2 Transmitting a Scout Image

When the scout image appears on the screen, touch ***X-mit S Image*** to transmit the image to a remote station. The image will be queued for transmission. When you are finished viewing, and you have returned home, The *Address Book* will automatically be displayed. (Please see Section 9 for directions on adding and deleting entries from your Address Book.) Choose a receiving station by touching the desired remote location. The selection will be highlighted in blue. Touch ***Select.***

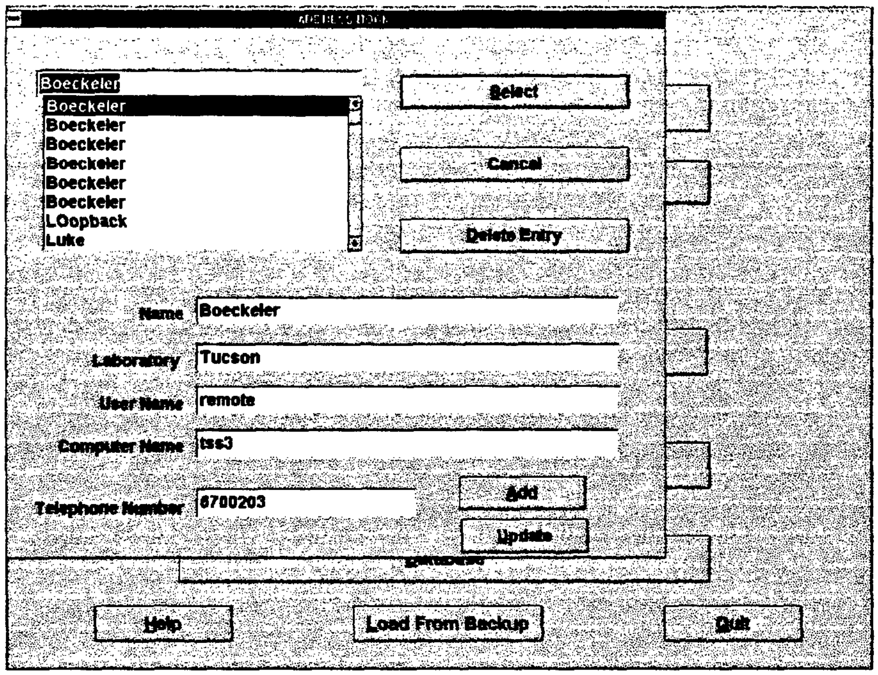


Fig. 12 - Example of the Address Book used to transmit images to different systems.

Once a location for transmission is specified, the user has the option of recording a message. A dialogue box will automatically appear asking, *"Do you want to record a message?"* If you wish to record a message, touch ***Yes***. Recording begins immediately showing a display of elapsed time. When finished recording, touch ***Stop***. In case an error was made during recording, touch ***Restart*** and record the message again. Touch ***Exit*** to end the recording session. A new dialogue box will appear requesting that the user click on ***Send*** to initiate transmission. The scout image will be transmitted to the chosen remote station. A progress bar will appear to document the connection with the remote station and the transmission of the image. Allow 5 to 6 minutes to complete transmission. [Note: Any green boxes which have been selected at the original location are not sent along with the scout image. The image is sent free of any overlay.]

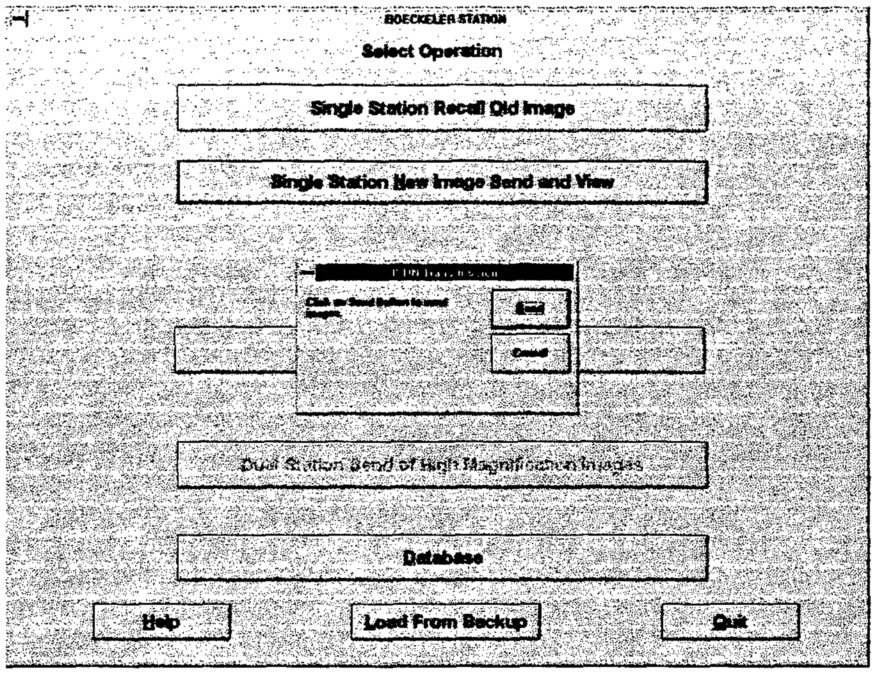


Fig. 13 - The dialogue box which documents transmission of an image to a remote station.

4.3 Receiving Requests for High Magnification Images

The remote station will receive the scout image, and areas of interest will be chosen for higher magnification. When the requests for these images are sent back to the original station, ***Dual Station Send of High Magnification Images*** will become highlighted on the *Main TSS Window*. Touch the button to begin filling the requests.

The *Request to Load Slide Window* appears, showing how many high magnification images were requested for any given slide number. If the correct slide is not already resting in the microscope carrier, load the slide into the carrier and touch ***O K***. The operator will be given the option of playing a message if one was recorded at the remote station.

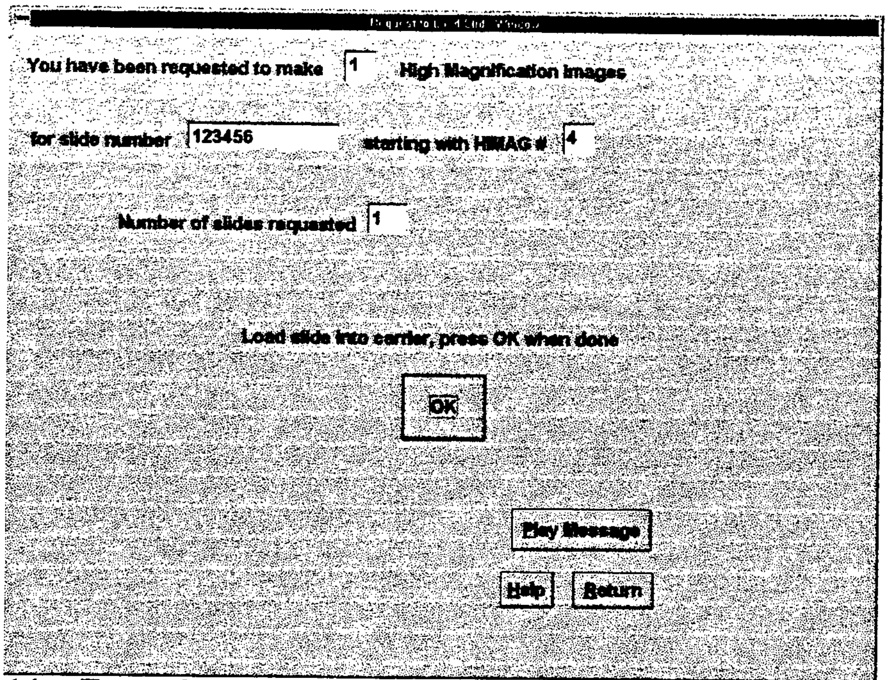


Fig. 14 - Example of the Request to Load Slide Window indicating how many high magnification images have been requested.

4.4 Sending High Magnification Images

The *High Magnification Image Window* will appear, showing the first X,Y location requested at the appropriate magnification. Focus the image using the up/down slider on the display screen. The operator has the option of changing the magnification, or "jumping" around to adjust the location of the image to be sent. However, it is a good idea to leave it at the original location and magnification since the image corresponds to the coordinates requested at the remote location.

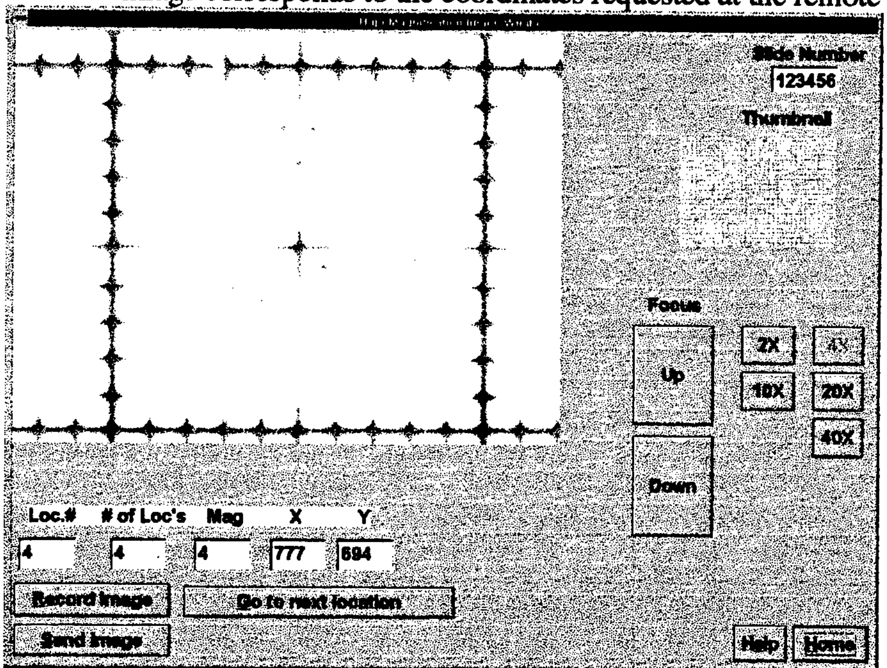


Fig. 15 - Example of the High Magnification Image screen. By touching the Send Image button, the image will be marked for transfer to another system.

Touch ***Send Image*** to mark the image for transmission. A dialogue box will appear offering the option to record a voice file. When recording is complete (See Section 4.2 for further recording instructions.), *"X-mit request added,"* will appear on the screen. Touch ***O K.***

To move to the next X,Y location requested, touch ***Go To Next Location.*** The microscope will automatically adjust to the appropriate location and magnification, but again the operator will be responsible for focusing the image.

Continue to fill all of the requests for high magnification images by following the steps in Section 4.4. When there are no more requests to fill, touch ***Go To Next Location*** to return to the *Recall Stored Images Window*. Touch ***Cancel*** to return to the *Main TSS Window*. At that time, all of the images captured will be sent to the remote station, and are saved at the operator's station. A progress bar will display the progress of the transmission.

4.5 Receiving a Diagnosis

After the remote station has received and the requested high magnification images have been viewed, any image with an important attached voice file can be sent back to the original sending station.

5. REMOTE MODE - RECEIVING

Remote Mode - Receiving is used when the workstation receives a scout or high-magnification image(s) from another workstation for evaluation. If a scout image is received, the examining pathologist selects areas of interest for high magnification and returns the request to the sending station. When the high-magnification request is returned, the pathologist may proceed with an evaluation.

5.1 Receiving a Scout Image

When a remote transmission of a scout image is completed, the ***Images Received*** button becomes highlighted on the *Main TSS* window. The transmission of an image takes approximately 5 minutes.

5.2 Viewing a Scout Image

Touch ***Dual Station Receiver***. This will bring up the *Recall Stored Images Screen.* From this screen, scroll down the list of scout images to the one or ones that have just been transmitted. The time, date, and place where the image originated will be displayed beneath the thumbnail of the image selected. Highlight the slide number to be viewed.

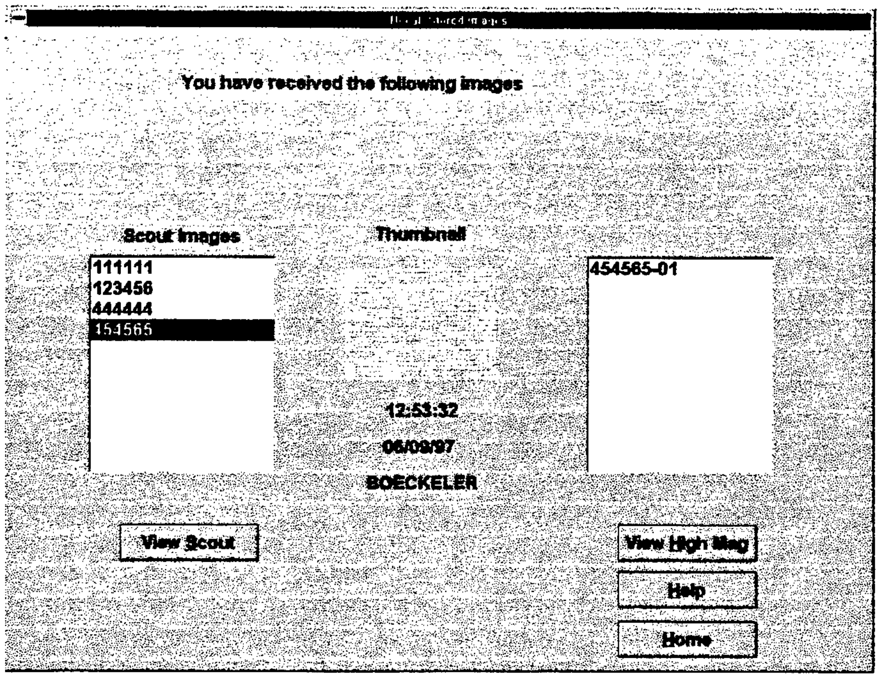


Fig. 16 - Example of the Recall Stored Images screen displaying images which have been transmitted to the system.

Touch ***View Scout Image.*** This will load the scout image and display it on the *Received Scout Image, Select HI-MAG* screen. This process will take approximately 1.5 minutes. The operator may listen to a recorded message regarding the slide by touching ***Play Message***

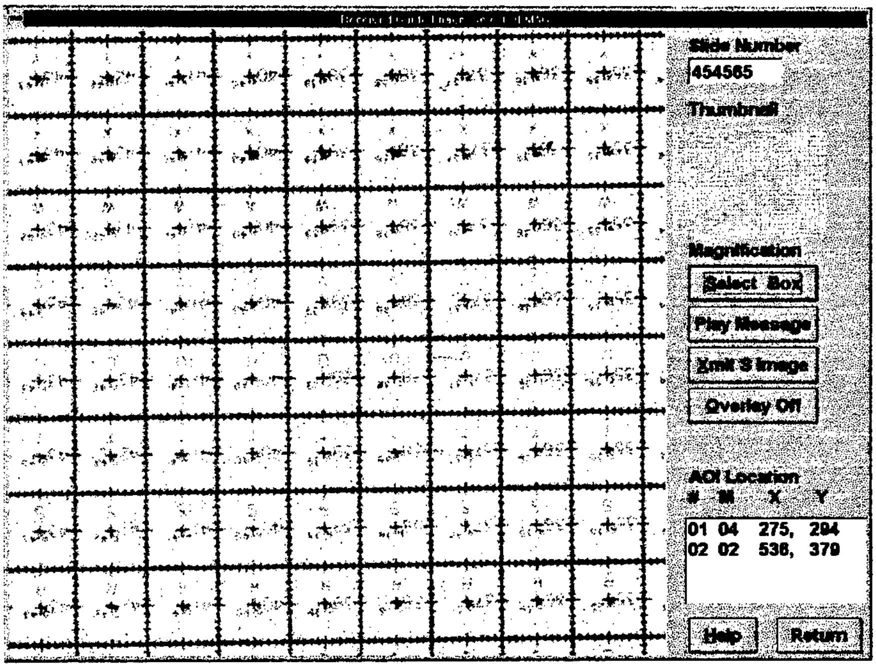


Fig. 17 - Example of the Received Scout Image, Select HI-MAG screen.

5.3 Marking Areas of Interest for Magnification

To mark areas of interest (AOI) for high-magnification examination touch ***Select Box***. A dialogue box will appear with magnification buttons for 2X, 4X, 10X, 20X, and 40X. Choose the desired magnification. . A small, square, black marker will appear on the screen in place of the white arrow marker. Touch the location on the scout image to be magnified. A black box surrounding the area to be magnified will appear. If the box needs to be moved to a more specific location, use the arrow keys on the keyboard to move left, right, up, or down. **Once the screen is touched again, the AOI box will turn green and become fixed in that location.** The size of the box surrounding the AOI will correlate with the desired magnification: a large box represents a low magnification while a small box represents a high magnification. In addition, the coordinates and the chosen magnification of the area will appear in the AOI Location Box in the lower right corner of the screen. The operator may continue to select locations and magnifications as desired.

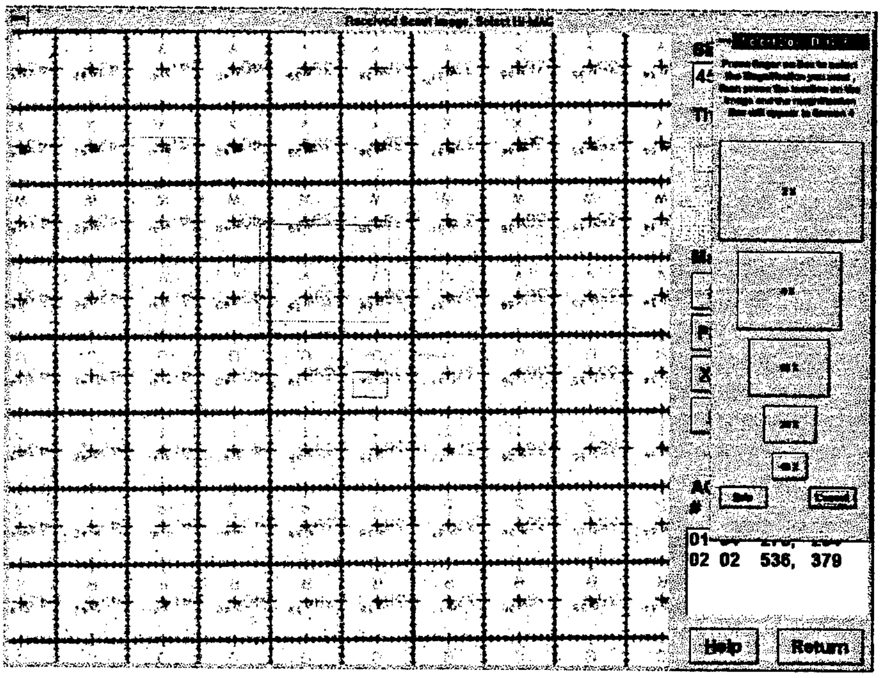


Fig. 18 - Example of the Selection Box used to mark areas of interest for higher magnification.

5.4 Sending a High-Magnification Request

When all areas to be magnified have been selected, touch ***Get HM Image***. The *Address Book* will appear. Highlight the receiving station and touch ***Select.*** The *Do You Wish to Record A Message?* box will appear. If you touch ***Yes***, recording will begin immediately. Record a message and touch ***Stop.*** To replace the message, touch ***Restart*** to begin recording again. Touch ***Stop*** and then ***Exit*** when finished. If you do not wish to record a message, touch ***No.*** A box stating *Request for HM Images Queued* will appear. Touch ***Okay*** to signal the station to prepare to send the images. The *Recall Sent Scout Images* screen will appear again. The operator **must exit** the *Recall Sent Scout Images* screen and return to the *Main TSS Window* to begin transmission of the requests for high-magnification images to the other station.

5.5 Receiving the High Magnification Images Requested

Images Received become highlighted when your high magnification requests are returned. Touch ***Dual Station Receiver*** to bring up the *Recall Stored Images* screen. Highlight the correct scout image slide number. The corresponding high magnification images should appear in the *High Magnification Box* on the right side of the screen.

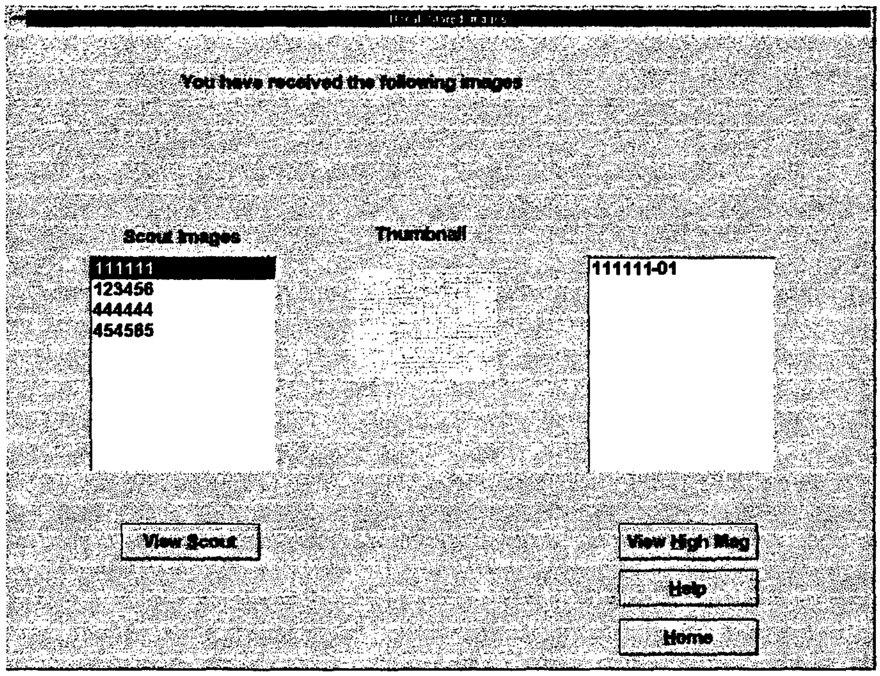


Fig. 19 - Example of the Recall Stored Images screen displaying the highlighted scout image and its corresponding high magnification images.

5.6 Viewing High Magnification Images

Select a high magnification image to examine and touch ***View High Mag***. The *Received High Magnification Image* screen will appear with the first requested high magnification image. A thumbnail of the scout image will appear in the upper right corner of the screen, with cross-hairs indicating which region is currently being examined at high magnification. The *Location #* will appear showing which high magnification image is being viewed (i.e. the first ,the second, or the third), as well as the X,Y coordinates and the magnification of that image. The *# of Loc's* will be displayed specifying how many high magnification images are present for the chosen scout image. Touch ***Play Message*** to hear any recorded message corresponding to the selected high magnification image.

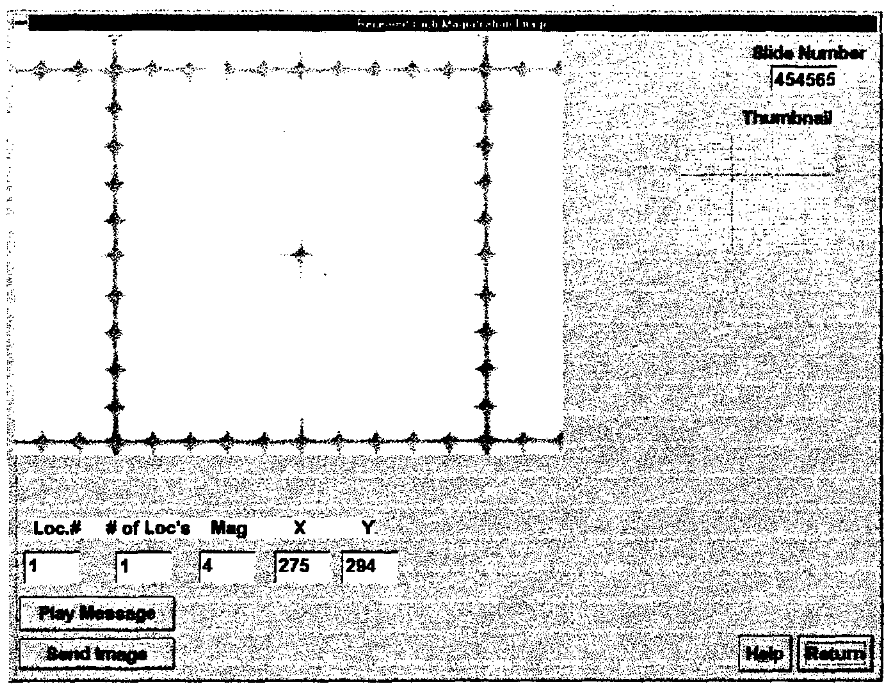


Fig. 20 - An example of the Received High Magnification Image screen.

To view the other high magnification images requested, return to the *Recall Stored Images* screen, and choose any high magnification image of interest. If no other high magnification images for the current scout image are available, the operator may go to another scout image and view its associated high magnification images also.

5.7 Recording a Diagnosis

To record any comments or diagnoses, touch ***Record Message.*** The *Record/Playback Message* window will appear on the screen and recording will being automatically. If during recording a mistake was made, touch ***Stop.*** To record another message, touch ***Restart.*** Touch ***Stop*** to end recording the message and ***Exit*** to remove the *Record/Playback Message* window.

6. NAVIGATING THE DATABASE

The database function is currently under construction. Please return to the main menu by touching ***Home.***

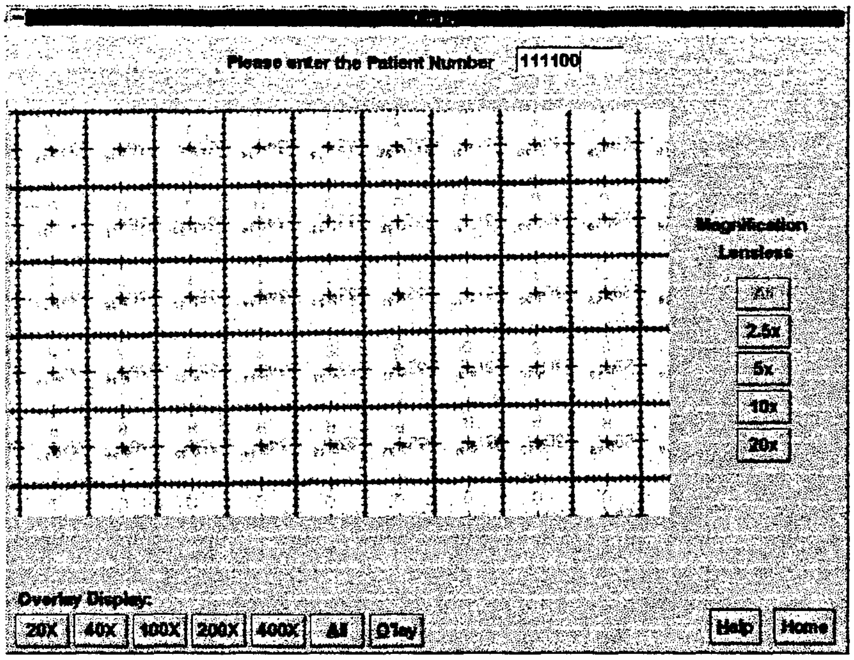


Fig. 21 - Example of the Database screen.

7. THE AUTOMATIC BACKUP FUNCTION

When a case is saved using the TSS, the information is stored on the hard drive. The amount of memory available for this function is quite limited. Therefore, an automatic backup function was added for the convenience of the user. This function requires that the user insert an empty 270MB Syquest cartridge into the Syquest drive onto which the cases will be stored. Ultimately, the cases are erased from the hard drive to free up space for new studies.

When the cases stored on the hard drive take up an amount of memory exceeding 200MB, a dialog box will appear prompting the user to insert a blank Syquest cartridge and back up the hard drive. The user may not disregard the prompt, or select only some cases to back up - **all cases will be backed up when the hard drive is full.** Insert the cartridge into the Syquest drive and touch ***O K*** to initiate the backup process. All case numbers will be displayed as they are transfered to disk. The cases will then be erased from the hard drive. Therefore, it would be helpful to record which cases were transferred to a particular disk so that they can be easily accessed in the future.

8. LOAD FROM BACKUP

Once cases are removed from the hard drive, they can not be accessed by simply pressing ***Single Station Recall Old Image*** at the main window. The case must first be loaded back onto the hard drive from a backup disk. Find and insert the Syquest cartridge with the desired case, and touch ***Load From Backup*** at the main window. A dialog box will appear, requesting the user to select which case they wish to load.

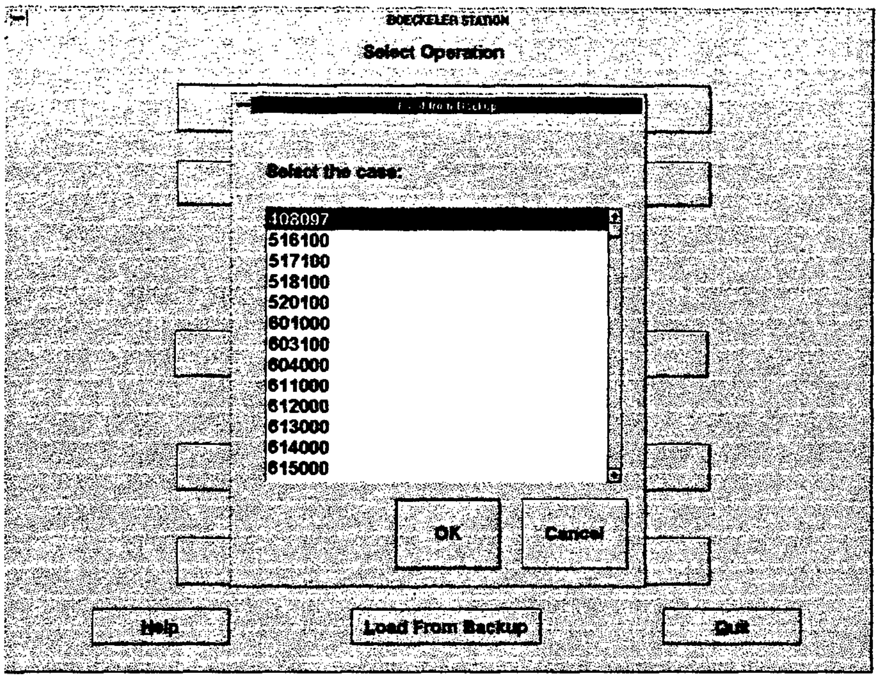


Fig. 22 - Choosing a case to load from a backup disk.

When the case number is highlighted, touch ***OK*** and the case will be copied onto the hard drive for viewing. When the user needs to access this case, simply touch ***Single Station Recall Old Image*** at the main window and follow the steps listed in section 3.7.

9. SHUTTING DOWN THE WORKSTATION

To shut down the workstation, first return to the *Main TSS Window* by touching ***Home*** from any window. Press ***Quit*** to end the session. From the dialogue box which appears, select ***Shutdown***. When prompted to shutdown, turn the green switch on the Powercell to the "Off" position. The entire workstation will be powered down. Replace the lens cap and microscope's vinyl dust cover, and wipe the touch screen clean of any fingerprints.

10. CARE AND MAINTENANCE

10.1 When not in Use

Always replace the lens cap on the condenser of the microscope which is located at the base of the microscope. Replace the microscope dust cover after each use.

10.2 Cleaning the Screen

Clean the touchscreen regularly with a household glass cleaner and paper towels to remove fingerprints and to maintain a clean, clear view of all images. Slightly dampen a towel and clean the touchscreen.

10.3 Cleaning the Microscope

To keep the microscope clean, wipe the stage with a soft cloth (silicon cloth is recommended). Avoid the use of any organic solvent on the painted and plastic surfaces of the

microscope. Dust can be brushed off the lenses, or wiped with lens tissue or a soft cotton cloth to remove fingerprints. It is safe to use pure alcohol if needed.

10.4 Do not Disassemble the Instruments

Remember, any damage caused by unauthorized use will void the warranty.

11. ADDING, EDITING, AND DELETING ENTRIES FROM ADDRESS BOOK

To add a name to your *Address Book*, simply touch on the **Name** box to move the cursor to that location. Type in the name of the person or system you wish to add. Press "Tab" on the keyboard to move to the next field, and continue to enter the information required: the laboratory at which they are located, the person's user name, the computer name, and the ISDN telephone number for that system. When this is finished, touch ***Add***.

If a person or system needs updated information, first highlight the name in the *Address Book* to bring up the detailed information below. Then simply change the information that needs updating. Touch ***Add***.

To delete a name from the *Address Book*, highlight the name and touch ***Delete Entry***. This will permanently delete the name from the address book.

12. TROUBLE-SHOOTING

Only the solutions to the problems covered in this section should be attempted by the user. If the answer to your problem is not listed below, please call Kensal Corporation.

12.1 Turning On the System / Logging On

The green button located on the Powercell is the only power switch to turn on before using the system. If you are having problems turning the system on, check that all electrical plugs are in their outlets and all connections between the computer and the microscope are secure.

Once the system has been turned on, the operator must log on to the program by entering the username and password assigned to the system by a Kensal technician. Press "Ctrl + Alt + Del" and the *Welcome* dialogue box should appear, prompting you to enter the information. When finished, press "Enter" on the keyboard, and wait about thirty seconds for the *Main TSS Window* to appear. If any problems arise, try restarting the system. If problems persist, contact Kensal Corporation.

12.2 Problems with the Touchscreen

If you are having trouble with the touchscreen, remember to press very firmly when you make a selection, and immediately release. The arrow that appears under your findertip will indicate exactly where you have touched on the screen. The system will make a beeping sound each time you touch the screen, however this does not necessarily indicate that you triggered the intended "button". If nothing happens, regardless of whether the system beeped, try touching the screen again, correcting your positioning to get the arrow onto the desired location.

If you are accustomed to using a mouse, you may request the technician to connect one to your system. When using a mouse, you simply position the arrow over the "button" you wish to press, and click the left button once, unless otherwise indicated. This may be especially helpful for using some of the scroll bars and smaller buttons. The touchscreen will still be activated should you wish to use both a mouse and the touchscreen.

12.3 Opening the TSS Program from Windows NT

If at any time the TSS program is exited, the system will display the Windows NT Program Manager on the desktop. It is not difficult to relaunch the program from this point. From the desktop, touch twice on the Program Manager icon in the bottom left corner of the screen. When the window entitled Program Manager - TSS/Administrator appears, click twice on the *Startup* icon at the bottom of the screen. When the *Startup* box appears, choose TSS from the box by clicking twice on the camera icon. This should return you to the *Main TSS Window.*

Sometimes the TSS application is not exited, but gets covered up by another open window. To return to the *Main TSS Window,* press and hold down the "Alt" button. A small window will appear in the center of the screen displaying the name of the program which is currently open. While holding the "Alt" button down, press and release the "Tab" button to toggle through all the open applications. Continue to press "Tab" until the name TSS appears. Release both the "Tab" and the "Alt" buttons to jump back to the TSS program.

If you are unable to return to the TSS program, restarting the computer and logging on from the beginning is a simple solution.

12.4 Problems Loading the Slide

Make sure the microscope slide is face up and the label is toward the back of the microscope. Gently glide the slide into position and secure it with the specimen holder. Sometimes using an object such as a pencil eraser to push the slide into place can be helpful when working in such a small area. See Figure 4 for proper slide orientation.

12.5 Problems with the Stage

If the stage is making any clicking or grinding noises during scanning, or if the slide is not entering the scan box when directed to create a scout image, immediately contact a technician. Do not attempt to rescan unless directed to do so by the technician.

12.6 Problems Producing a Scout Image

If the scout image on the screen appears white, check that the slide is facing up, and that it is placed in the specimen holder with the label towards the *back* of the microscope. The lensless scanner only views the forward section of the slide, so if a slide is put in place backwards the resulting scout image is of the label instead of the tissue.

If the image appears washed out, blurred, or streaked, check that the slide is laying flat on the stage, well secured by the specimen holder, and rescan the specimen. It is often helpful to run a few trial scans to allow the system to warm up. Fingerprints and dust on the coverslip will blur the image, so it is important to clean the slide well before scanning. The slide can be wiped clean with a soft cloth or tissue, and dust can be removed using a can of condensed air. If there are still problems with the image quality, contact a technician.

12.7 Problems Viewing a High Magnification Image

If you are trying to look at an area of interest, but no high magnification image appears, check that the lens cap has been removed from the microscope. Be sure that the appropriate slide is secured on the stage. The objective lens on the microscope should be locked into position, not resting between notches. If there is still no image, contact a technician.

When an image appears it is often very blurry. If you are having trouble focusing the image with the focus button on screen, keep in mind that the process occurs slowly. The focus

buttons correspond to the fine focus knob on the microscope, so you must hold the button in the up or down position while the focus slowly adjusts. If the image still does not appear focused, use the focus knob on the microscope itself. This is the large black knob located on the right side of the microscope, near the back. There are two portions to it: the part closest to the microscope body adjusts the coarse focus while the outer part adjusts the fine focus.

If after adjusting the focus the high magnification image still appears cloudy or dark (especially at a 40X magnification), you can manually adjust the filters to allow more or less light to pass through. The filters are located on the right side of the microscope base, towards the back. Adjusting any of the six filters will change the light intensity of the image, making it lighter or darker depending on if you push in or let out the buttons. Find a light level that allows you to see the most detail before you capture the image since light adjustments can not be made on a fixed image.

12.8 Problems Sending or Receiving an Image

If there is a problem transmitting an image, an error message will occur. Typically it takes up to a minute for this message to appear because the system will make several attempts to complete the transfer before recognizing the error.

In order to send or receive an image, **both** systems must be turned on. If a transmission error message appears, check with the receiving end to be sure their system is turned on. If the problem continues, next check in the address book that the ISDN telephone number is correct for the remote system.

Your phone company is responsible for setting up and providing the service for your ISDN lines. If a transmission error occurs, please check with your local phone company regarding the status of your ISDN lines. Contact a manufacturer technician if you are still unable to send or receive images at your workstation.

12.9 Problems Choosing an Area Of Interest (AOI)

If you are having problems getting a box around the exact location to be magnified on a scout image, remember that as long as your finger/pointer remains on the screen, the black marker will follow your touch. If, when you take your finger/pointer off the screen, the marker jumps, simply adjust its position with the arrow keys on the keyboard. The box will remain black indicating that it can still be moved. Do not try to readjust it with you finger/pointer. As soon as you touch the screen again with your finger/pointer, the box will turn green and become fixed in that position.

12.10 Problems Recording/Playing a Message

If you are having problems recording a message, make sure that the microphone is turned on. There is a switch on top of the microphone; check to see that it is in the "On" position. Try recording the message. If you are still unable to record a message, contact a manufacturer technician.

If you are having problems hearing a recorded message and you know that there is in a fact a message to be heard, check that the speakers are turned on. There is a volume knob on one of the speakers that can be turned up to increase the volume.

12.11 Using the Help Index

The operator may press ***Help*** from any screen to access the on-line Help Index. By touching any of the green, underlined commands in the menu, the corresponding help file will appear on screen. To return to the help menu, touch ***Back***, found on the gray menu bar at the top of the screen. Browse the help application by pressing on the ***Search*** option. To exit *Help*, touch ***File*** on the white menu bar, and when the menu drops down, touch ***Exit*** to return to the *Main TSS Window.*

13. GLOSSARY OF TERMS

AOI	"Area of Interest"; regions which a pathologist feels might be helpful in making a diagnosis; usually marked regions on the scout image which are to made into high magnification images
CCD Camera	"Charged Couple Device" camera; a picture-taking device which records lensed microscopic images
Computer tower	The multimedia computer including the hard drive of the TSS
Cross-hairs	Two lines criss-crossed showing the location being viewed on a scout image
Field Finder	The sample slide that comes with the TSS
Scout Image	The lensless image of a full-coverslip glass slide
HM	"High Magnification" image
ISDN Lines	"Integrated Service Digital Network" lines; high-speed telephone lines
JPEG	"Joint Photographers Expert Group"; a standard for image compression
Jumping	A process to scroll and view images by; any touched region of an image will "jump" to the center of the screen
Lensed Scanning	Microscopic images produced with a camera, such as the CCD camera
Lensless Microscopy	The application of a microscope which requires no lenses; permits rapid, full-coverslip imaging at a very high resolution of microscopic slides.
Local Mode	Referred to when the TSS is used as a stand-alone system
Real-time	The actual time in which a physical process takes place; live transmission
Remote Mode	Referred to when the TSS is used for sending and receiving images in conjunction with a second workstation

Resolution	The process or capability of making distinguishable the individual parts of closely adjacent optical images.
Specimen Holder	The latch that holds the glass slide in place on the microscope stage
Static Image	Fixed in place; stationary
TSS	"Telemedicine Support System"; a telecommunications system for pathology that integrates lensless imaging with lensed imaging.
Thumbnail	A small representation of the entire scout image

APPENDIX B

TPW DESIGN DOCUMENT

Tele-Pathology Workstation Interface and Object Design

Prepared by:
Jay A. Nance

Date:
22 February 1997

1 INTRODUCTION

This design document has been prepared for the perusal and use of the software implementation team. This design contains all information needed to fully develop the Tele-Pathology Workstation Software. It is understood that some items may change and some items could simply not be identified at the time of writing this document. In these cases the resolution will happen during development to the satisfaction of Kensal.

The Tele-Pathology Workstation is designed to be utilized by multiple pathologists working at various locations around the world. At the time of this writing the usage will be limited to pathologists communicating via a network of ISDN Connections. The system is designed to be a communal workstation (multiple users at one site) with a built in camera unit, 1280x1024 monitor, running on an Apple PowerMac 9500. The software will be developed using Symantec C++ 8.0 (or later) with the Think Class Library (TCL).

The system is built on a main window which is used for the majority of the interface. Dialog boxes and floating windows supplement the interface where needed. See Section 2 for a detailed discussion of each interface object. The system will operate in what has been termed "modes". Each mode is defined in section 3 and includes a flow chart which outlines the operation of each course of action (see below for a brief introduction to the modes). Section 4 describes the C++ Class Objects which will be developed during the implementation phase. A complete identification of the files needed to implement the system is described in section 5.

The primary product of the Tele-Pathology Workstation is a set of images of which there are two types. The first type is the guide image of which there will never be more than one per slide analysis session and it will always be a scan of the entire slide at 4000x8000 pixels. The second type will be the high magnification images which will have many occurrences at various locations and magnifications throughout the slide. See section 5 for a discussion of these and other file types.

One of the more confusing aspects of the Tele-Pathology Workstation is the concept of modes. These modes are necessary so that when the user is performing a given function that the system operates in a manner unique and consistent to that function. A mode describes an environment in which the user operates to perform the aforementioned function. It is important to note that rarely will a particular user operate in all of the modes. Typically the user will operate strictly as either the remote expert or the local technologist. The modes are identified as:

Dual Station

<u>Generate Guide and Transmit</u>
The local user will generate a guide image and transmit to the remote expert a request for analysis of the slide.

<u>Request High Magnification Images</u>
The Remote expert will review the guide and transmit back to the local user a request for high magnification images at specific locations and magnifications.

<u>Generate High Magnification Images</u>
The local user in turn fulfills the high magnification request and returns to the expert the specified images.

<u>Diagnose From Images</u>
After having received all images necessary the remote expert will dictate various voice message diagnoses and transmit to the local user the request for transcription.

<u>Transcribe Diagnosis</u>
The local user performs the transcription task for the slide analysis session which completes the Dual station analysis.

Chat
The chat is a mode modifier in that a chat can be requested with virtually any user at any time from any mode. During chat the users involved will be essentially sharing screens and communicating over the voice channel as if on the telephone.

Single Station

Examination
If the user generates a guide and then goes on to analyze the slide at high magnification they will be automatically placed in examination mode.

Review Overlays and Diagnose
To conclude the Examination the user leaves voice messages at various locations and submits the session for transcription.

Database
There will be available to the user a selection of databases through which browsing is possible. These databases could be the users own cases, Kensal supplied cases on CD-ROM, or various Internet sites.

2 INTERFACE LAYOUTS

test The following figures are the proposed windows and dialogs making up the Tele-Pathology Workstation. Accompanying each interface is a set of layout objects which are identified in the subsequent text. Each object is then discussed in enough detail to describe its functionality and use. This is not intended to give the reader a comprehensive understanding of the operation of the system. For operations see section 3.

2.1 Main Window

The Main Window of the Tele-Pathology system performs all of the vital functions for the software interface. All other windows and dialogs are supplementary. When the user launches the system the main window will be displayed. The Tele-Pathology is seen as an executive type system. Executive systems have the primary requirement of ease of use. In the Tele-Pathology system this will be accomplished in the following ways.

1. No Pull-down Menu
   All of the options and commands will appear on the screen itself and give the user as much information as possible. The objects will also respond to the current mode and environment by dimming or updating their names as described below.
2. A Single Main Window
   In order to simplify the system the main window will offer at least a starting point for every function which the user needs to perform.
3. Main Window consumes entire screen
   The main window will also sense the screen size on launch and resize to fill the entire screen. This reduces clutter on screen and focuses attention on the Tele-pathology system. All window objects will maintain their relative positions and actual sizes for any screen size except for the image port which will relatively maintain both position and size.

Note that the figure below shows all controls on the main window. This is not meant to represent the window as it might actually be seen during any actual mode, but rather to illustrate the functional options in the window.

Also note that some of the buttons change names in accordance with the mode of operation. The buttons which can have different names and functions are:

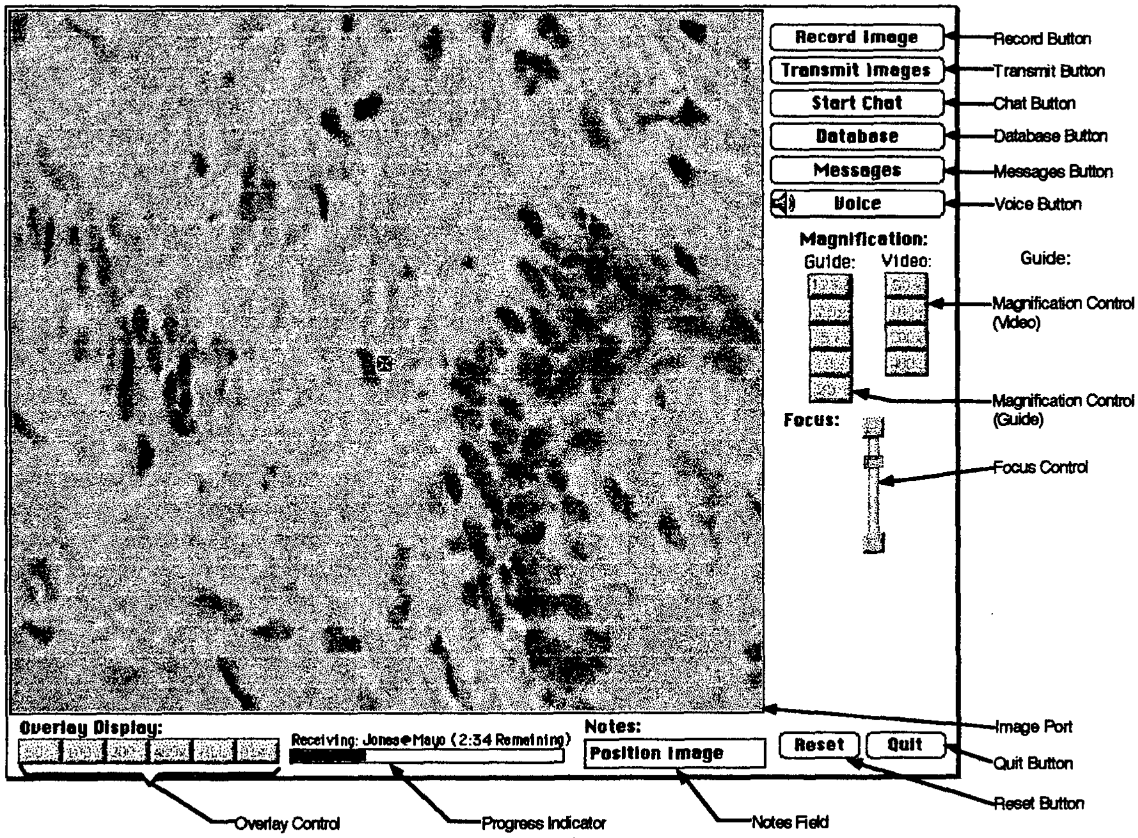
Button Name	Section	Can Read...
Chat	2.1.1	Start Chat, Resume Chat, End Chat
Record	2.1.12	Record Guide, Record Location, Record Image, Next Image
Transmit	2.1.14	Transmit Guide, Transmit Locations, Transmit Images, Transmit Diagnosis

Throughout the following text the controls will be describe in terms of various states of functionality. It is important to note that this refers to the slider and controls as well as the buttons. These states consist of:

Disabled	This will be displayed as a dimmed control. The dimming of the control indicates to the user that the control is currently not functional but that it will be once a set of criteria are met.
Enabled	The enabled control is fully functional.
Available	The available control is fully functional.
Unavailable	The unavailable control is simply not on the screen. This is most commonly achieved by hiding the control. This state is used in modes which will not ever use the control in question.

Deselected Several of the controls utilize a selection mechanism. At various times it is necessary to set all of the selections off. This is achieved by deselecting the control. The control remains fully functional.

Read Only Some of the controls are used not only to get input from the user but also to give the user information. Sometimes these controls are only for displaying the information. In such states the controls are read only.



Main Tele-Pathology Window showing all controls

The following lists each of the screen objects and their functions.

2.1.1 Chat Button

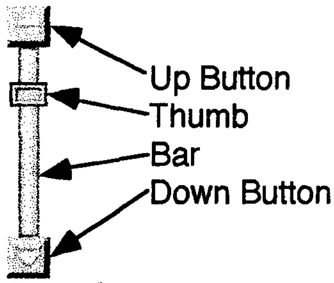
The chat button will allow the user to enter a chat. After pressing the chat button the transmit dialog will be displayed and the available chat users will be given as choices in the list. An available chat user is one who is a "direct" connection. Once the connection has been established the users will share the main image display, magnification controls, focus control, overlay Control, and Mouse. The users will also have a direct sound connection over the standard sound in and out ports. When requested to enter a chat the user will be prompted with a dialog and requested to accept or deny the chat. After connection the chat button will read "End Chat".

2.1.2 Database Button

The database button is designed to allow the user to select previously saved cases by entering search criteria for: slide number, analysis start date, analysis completion date, organization MD, medical institution, organ, disease, and/or SNOMED code.

2.1.3 Focus Control

The Focus control is made up of 4 components:



Focus Control

Each of the components work in conjunction with the others and all are live controls (meaning that they send the focus command to the camera, the camera responds and the image is updated even while the mouse button is still down). The up and down buttons will move the focus up and down (while updating the thumb position) by a scaled increment achievable by the camera. The bar will move the thumb to the position on which it is clicked and focus to that relative increment. The thumb will adjust the focus to it's incremental position while it is being drug along the bar. The number of focus pulses which the control sends to the camera is inversely proportional to the magnification of the camera. At 400x the focus control will send 1 pulse per increment. At 50x the control sends many pulses ("many" to be defined by the focus stepper motor driver).

The focus control can only be used when the slide is actually on the table. Therefore the focus control will be dimmed whenever the system is being used by the remote user or when the local user is simply viewing stored images or performing other non-image control functions.

2.1.4 Image Port

The image port will always contain the currently selected image at one of the acceptable magnifications. For a guide image creation the image port will contain the guide image at All (1.25x), 2.5x, 5x, 10x or 20x Magnification. During High Magnification viewing the image will be displayed at the selected magnification (50x, 100x, 200x, or 400x). The user will be able to jump the image using the point and click method. This method functions by clicking in the image at a point to which the user wants the center of the image to move. The center of the image will be indicated by a cross-hair type icon which remains stationary as seen above ().

2.1.5 Magnification Control (Guide)

During any session the guide image is available. The guide magnification control always operates on the guide image and performs software magnifications only. The control button consist of "All" (1.25x), "2.5x", "5x", "10x", and "20x." The following chart illustrates the image sizes at the various magnifications.

Guide Magnification	Image Size
All (1.25x)	1000x500
2.5x	2000x1000
5x	4000x2000
10x	8000x4000
20x	16000x8000

2.1.6 Magnification Control (Video)

The video magnification control will be activated during high magnification request sessions. A single station session where the local user is also the expert will behave in functionally the same manner. At all other times it will be disabled but will display the current magnification. During high magnification request the magnification control will not actually magnify the control as the actual image will not be available since the magnification is being set by the remote user. Instead the guide image view area will display a dotted rectangle showing the size of the image for the selected magnification. The magnification rectangle will always be centered on the view so that after the magnification value is chosen the user can fine tune the position before pressing record button. During high magnification creation the video magnification control will drive the camera to set the requested magnification and display that image to the main image display.

2.1.7 Messages Button

All incoming telecommunications will be summarized in the message list in the messages dialog. To access the messages dialog the user will press the messages button. The messages button will only be available if messages exist which have not been responded to. Messages will remain in the list until the request has been responded to and the user quits the application. Once the message has been responded to it will disappear from the list. A permanent ASCII format messages file will be stored which logs all incoming and outgoing messages for accounting purposes.

2.1.8 Notes Field

The notes field will be updated on a active basis to instruct the user where they are, what is coming next, etc. The particular items to be included in this item are as yet unresolved and will be determined during development.

2.1.9 Overlay Control

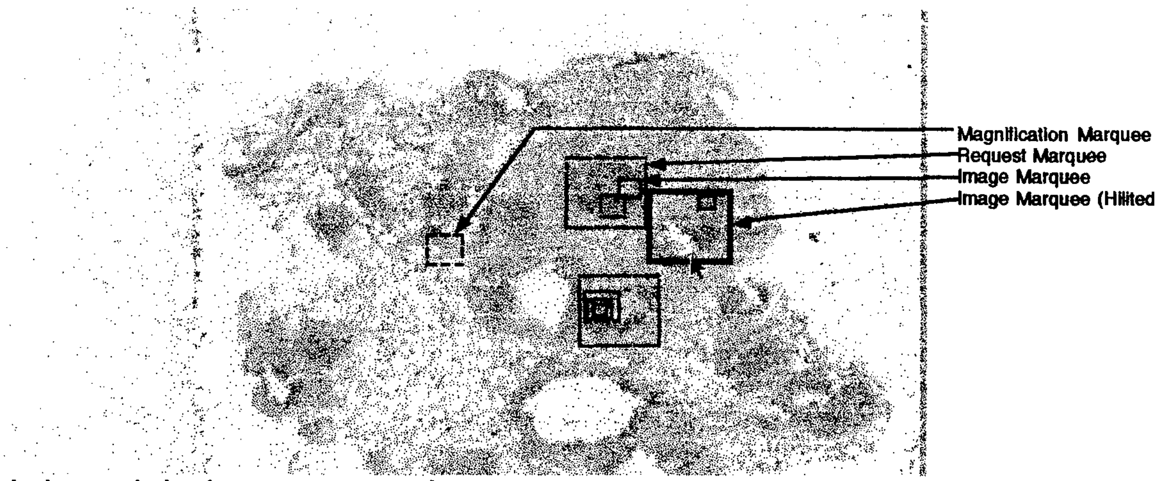
To show or hide the Overlay for a specific magnification the user will click on one or more of the Overlay control buttons. If all of the buttons get selected the "All" button will automatically depress. Conversely, If the user presses the "All" button all of the overlay magnification buttons will automatically depress. The "O'lay" button will toggle between no overlay and overlay mode. In examination mode the overlay will be seen as a series of colored rectangles (one color for each magnification) for each location which has visited by the user.

The locations at which the user has recorded an image will by displayed as black rectangles in all modes. These black rectangles shall be known as image marquees. As the user moves the cursor over these image marquees the black line will "hilite." This will allow the user to easily see which image will be selected when the user clicks. The image marquees will be arranged on the screen in such a way so as to make all images selectable.

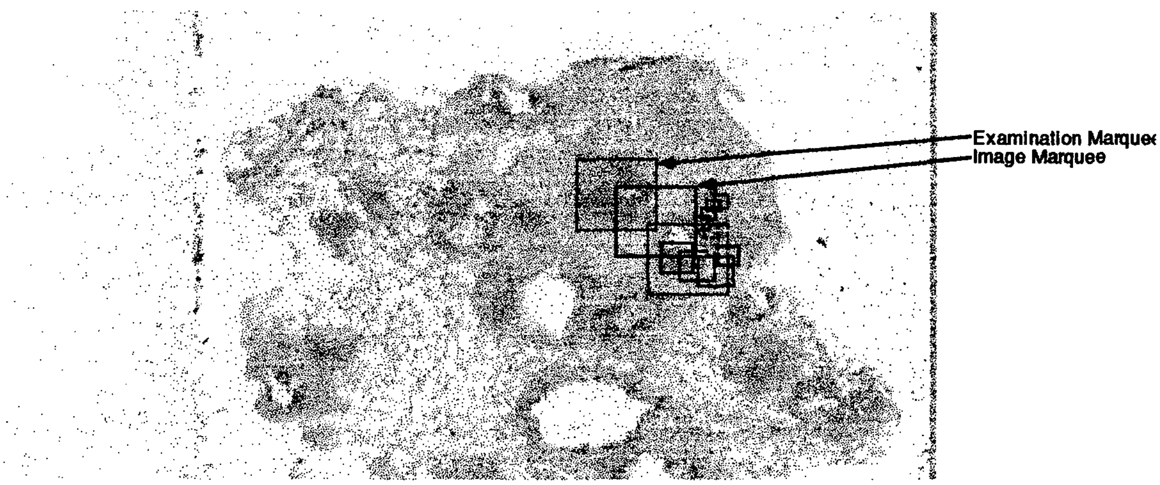
During image creation mode the locations will be indicated with black cross hairs. However only the cross hair for the current location will be displayed at any given time. These cross hairs will be known as request crosses. See section 2.8 for an example of the request cross

During examination mode when the user simply visits a site the marquees will be solid color rectangles without numbers. These color marquees shall be known as examination marquees.

During image request mode the locations will be indicated as dash line boxes. These dashed lines will be known as request marquees. Before the requested location has been recorded the image port will display a dashed line rectangle indicating the relative size of the image at the current high magnification factor. This dashed line box will use the "marching ant" mechanism for indicating the current selection and shall be known as the magnification marquee.



An image during image request mode



An image during examination mode

The overlays which are displayed on screen will be of various sizes depending upon the guide magnification. The following table illustrate the marquee size according to the image magnification and guide magnification. The table also shows the examination marquee colors.

Optical Magnification	Color	Guide Magnification 1.25x	2.5x	5x	10x	20x
400x	Red	5x4	9x8	18x15	32x28	64x56
200x	Yellow	10x8	18x16	35x30	65x56	130x110
100x	Magenta	20x16	36x32	70x60	130x110	324x228
50x	Cyan	40x32	72x64	175x50	324x228	648x456

2.1.10 Progress Indicator

The progress indicator will display the current status of any background operations. The text associated with the indicator will display the action (sending or receiving), remote site, and remaining time. When no activities are taking place in the background the progress indicator will not be visible.

2.1.11 Quit Button

To quit the application and return to the finder click the quit button. Any unsaved or uncommitted work will be requested to be saved or committed before exiting.

2.1.12 Record Button

During a guide image creation session the user will have loaded the slide. The record button (titled "Record Guide") will store the image to file in guide image format.

During a high magnification request session the user will position the view to the approximate location of interest and chose a magnification from the video magnification control. The record button (titled "Record Location") will store the location and magnification to a file to be transmitted to the local site. If the user is viewing previously recorded high magnification images they can step through the images by pressing the record button (titled "Next Image").

During high magnification creation the user will have chosen the image position within a selected location, focused the camera, and fine tuned the scroll to the area of interest. After having composed the image the user will save the image by pressing the record button (titled "Record Image"). The actual location will be saved with the image in addition to saving the requested location which may be different. After saving the image the record button will be dimmed until the user chooses the next location. The voice button will still be active allowing the user to store a voice message if desired. See section 3.3.

2.1.13 Reset Button

To reset the current mode and return to non-mode the user may choose the reset button. Any unsaved or uncommitted work will be requested to be saved or committed before resetting.

2.1.14 Transmit Button

The transmit button will be used initially for sending the guide image to the remote expert in guide image creation mode (button title is "Transmit Guide"). The remote user will also use the transmit button to send the location request file back to the local user during high magnification image request mode (button title is "Transmit Locations"). After all images have been recorded and voice messages stored in high magnification image creation mode, the local user will transmit the images back to the remote user by pressing the transmit button (button title is "Transmit Images"). After a analysis is complete and the remote user has dictated a diagnosis, in diagnose from images mode, the transmit button is used to send the diagnosis voice file back to the local user for transcription (button title is "Transmit Diagnosis").

When the user is in any mode other than guide image creation the transmit button will transmit to the user who requested the images, otherwise the button will invoke the transmit dialog and request the address for transmission. The transmit button will pass the image/voice file names and remote user addresses off to the communication object which will operate in it's own communication task environment.

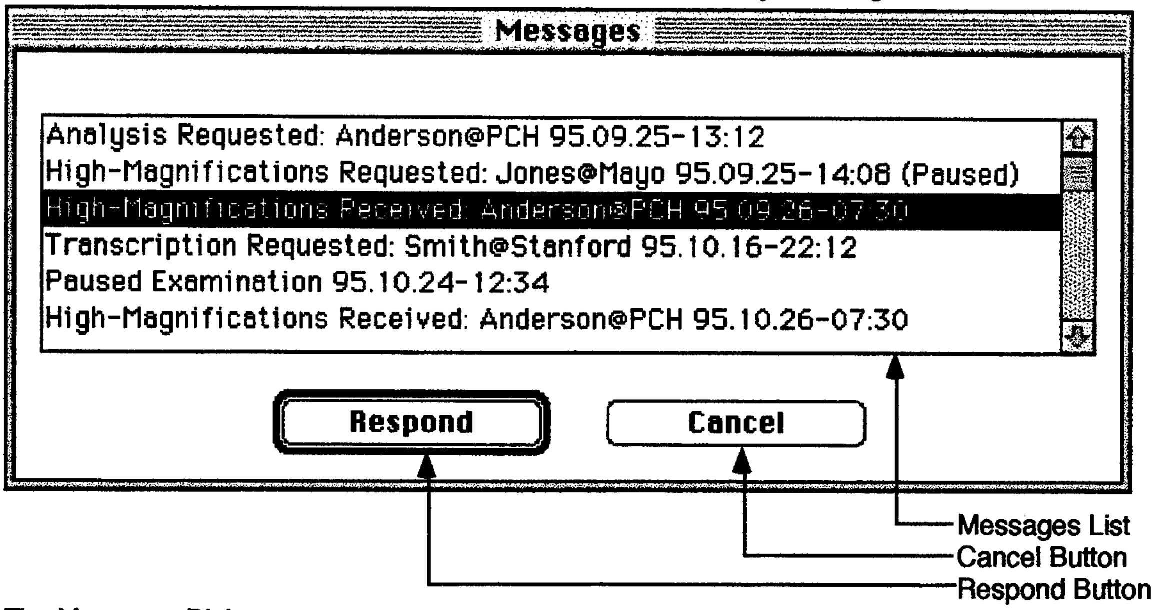
During a transmit the progress indicator will contain an indication of the transmission status and progress. This will occur at both the local and remote sites.

2.1.15 Voice Button

Messages can either be played or recorded from the voice button. The voice button will always invoke the voice dialog. If the users current location has a voice file associated with it the button will display an icon indicating that voice is available for play. The guide image itself is a valid location for voice to be associated. If the user is in a mode which does not allow the recording of voice (I.E. database or transcription) and the location does not have voice associated and neither does the guide, then the voice button will be disabled. For a complete discussion of the voice dialog see section 2.5.

2.2 Messages Dialog

In order to choose an action item and enter one of the tele-pathology response modes the user must select an item from the messages list in the messages dialog.



The Messages Dialog

2.2.1 Cancel Button

The user can abort the selection of a request by pressing the cancel button.

2.2.2 Messages List

All received requests as well as work items on pause will appear in the messages list. By default the first item will be selected. The user can select an item and press the respond button or simply double click the item.

2.2.3 Respond Button

After selecting an item in the messages list the respond button is enabled and the user can enter the mode required by the list item by clicking on the respond button.

2.3 Database Search Dialog

The search dialog is accessed when the user presses the database button from the main window or when the search button is pressed in the saved images dialog.

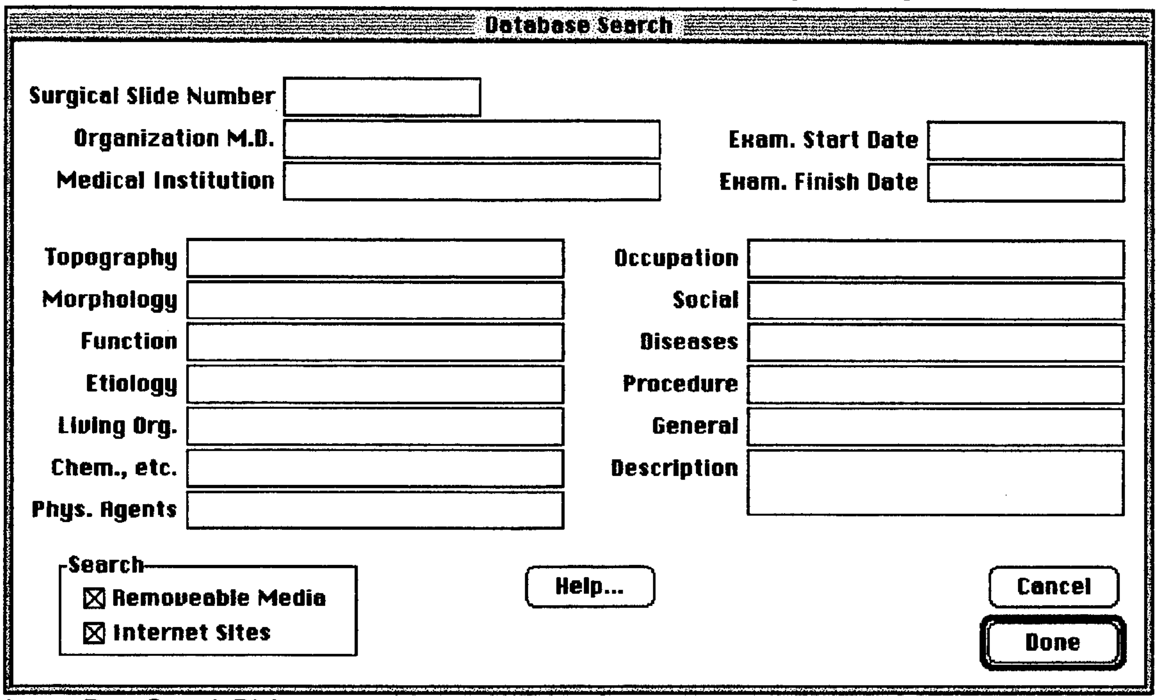


Image Data Search Dialog

2.3.1 Cancel Button

To exit without executing the search press the cancel button.

2.3.2 Databases Button

The user will be able to choose the databases which will be interrogated to find the requested images via the databases button.

2.3.3 Done Button

To execute the search press the done button.

2.3.4 Entry Fields

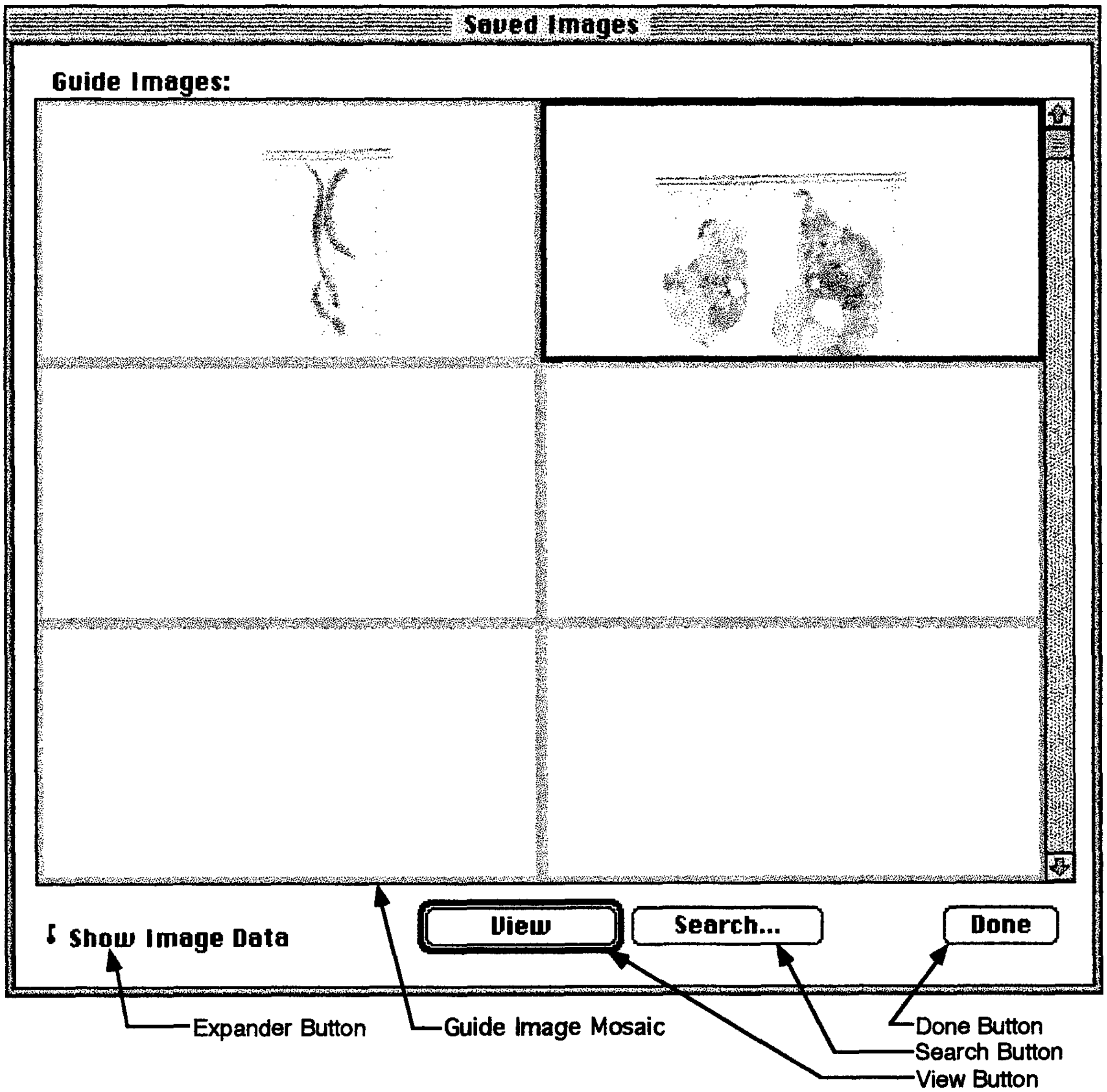
Enter all search criteria in the entry fields. use wild cards as necessary ('*' = multi char, '?'= single char, []= single position string). If no criteria is entered all saved images will be returned. All text searches will be non case sensitive.

2.3.5 Help Button

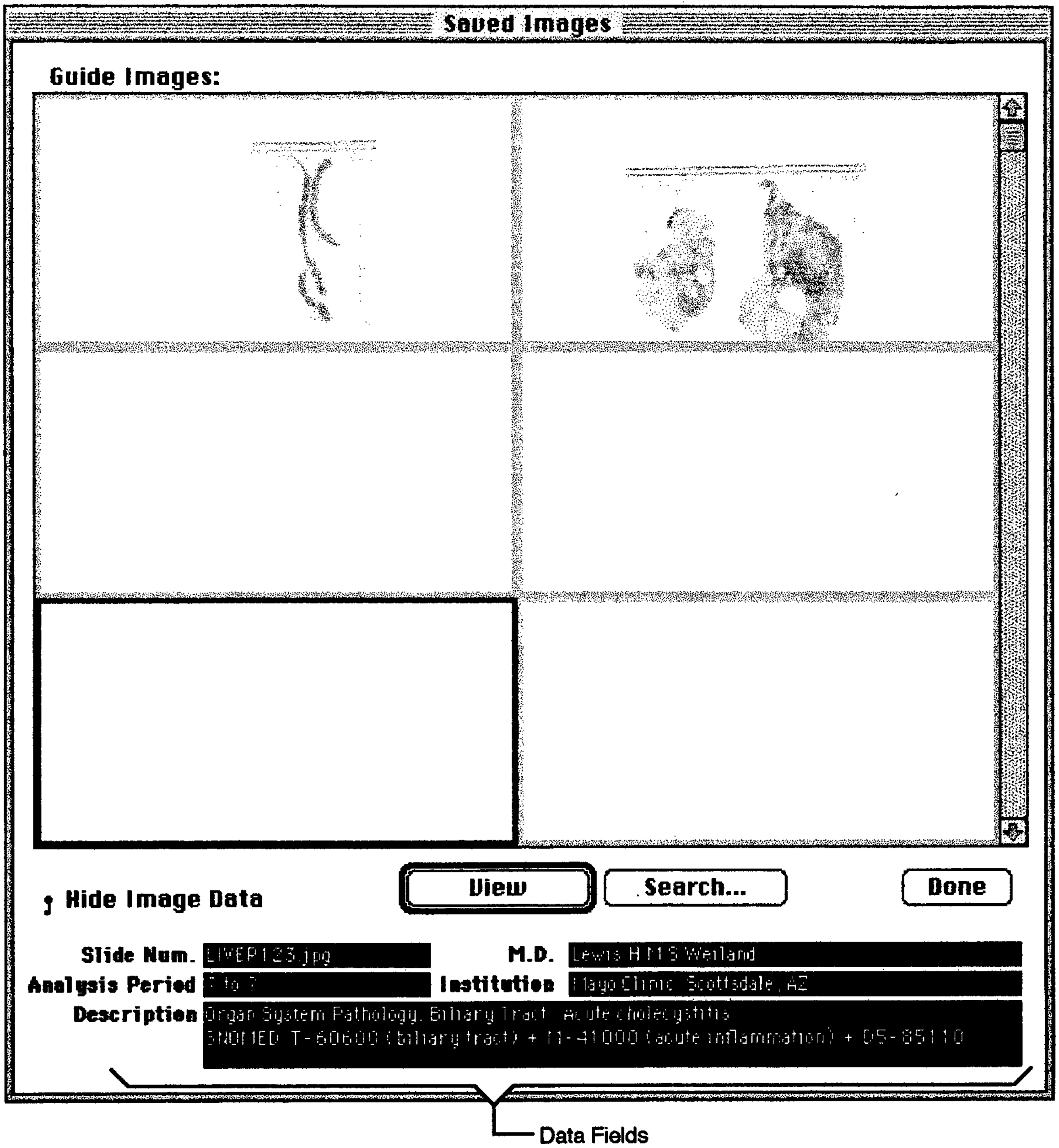
The help button will invoke a help window which will explain the use of wild cards and search criteria.

2.4 Saved Images Dialog

The saved images dialog is a moveable modal dialog which requires a response before the user can continue. Any of the buttons identified below will release the dialog and return to the main window.



Saved Images Retrieval Dialog



Saved Images Dialog in Expanded Mode

2.4.1 Data Fields

The data fields display the image data for the selected image. These fields are view only and will automatically change when the user selects a different guide image.

2.4.2 Done Button

To exit the dialog without selecting a saved image press the done button. This will be the default button if no image is selected.

2.4.3 Expander Button

To view the image data associated with the selected image the user can press the expander button which will resize the window and display the data. When expanded the window can be contracted by pressing the expander button again. The text will change in accordance with the mode of the dialog. The text will also be part of the button (i.e. it "wants clicks").

2.4.4 Guide Image Mosaic

A thumbnail representation of the guide images selected via the search will be displayed in the guide image mosaic. The guide image will also show the high magnification locations as marquees or dots.

2.4.5 Search Button

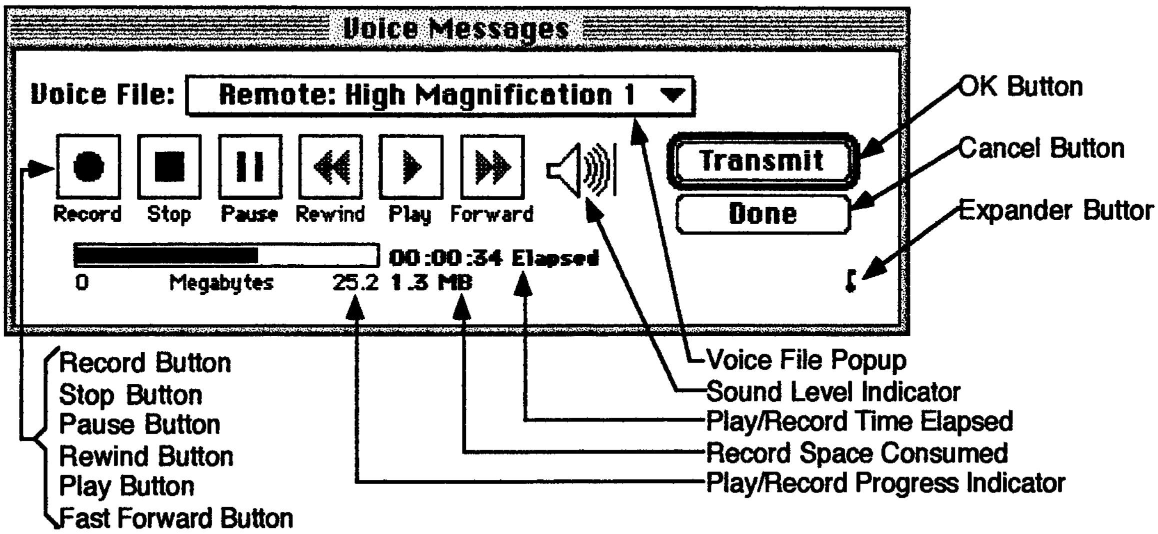
The search button will invoke the database search dialog allowing the user to select a new set of images.

2.4.6 View Button

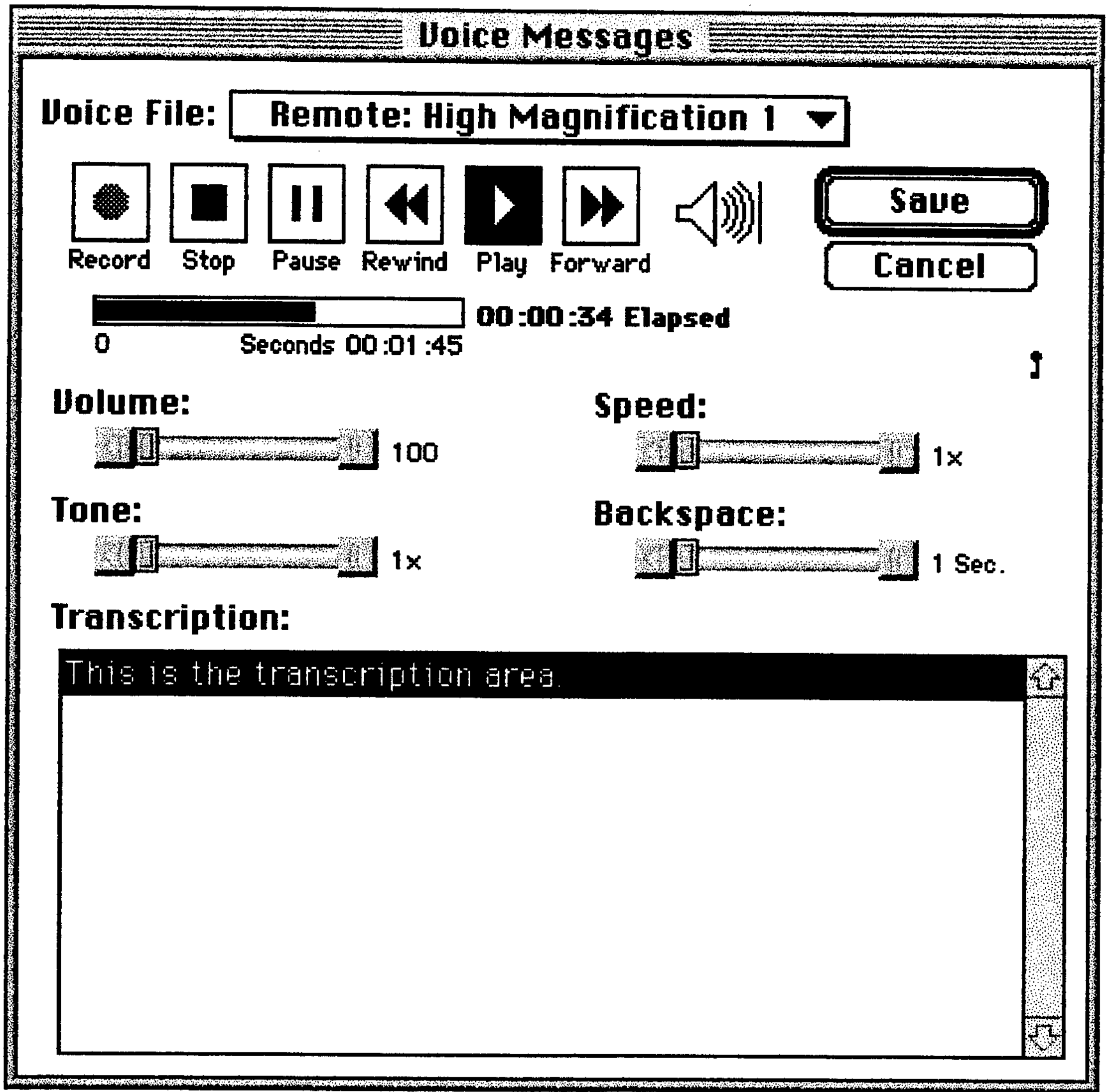
To view an image and its associated high magnification images press the view button. This will release the dialog and return to the main window. The high magnification images stored for the current guide image will be available through the marquees in the main window.

2.5 Voice Dialog

The voice dialog is a moveable modal dialog. The voice that is being played or recorded through the dialog will be associated with the image or location displayed in the main window. This dialog is modeled after the sound addition dialog in the standard Macintosh operating system. Consistency between the standard dialog and the sound dialog in the Pathology Workstation should be maximized. The voice dialog will be utilized by the transcriptionist in expanded mode. The transcriptionist will typically have a foot treadle device which will be connected via the ADB or serial port and will control the fast forward, rewind, pause and play functions of this dialog.



Voice Dialog (Record Mode)



Expanded Voice Dialog (Play Mode)

2.5.1 Backspace Control

The backspace control sets the distance in seconds that will be set behind the current play position when the rewind control is pressed. The control range is from 1 to 10 seconds in 1 second increments. The default setting is 1.

2.5.2 Cancel Button

To release the dialog without saving the sound message to the image file press the cancel button.

2.5.3 Expander Button

The expander button is available during diagnosis transcription. This button expands the voice dialog so that the transcription field can be typed into while the dictated diagnosis is being played.

2.5.4 Fast Forward Button

Fast forward the voice play. This control increments the speed control by one each time it is pressed.

2.5.5 Pause Button

The pause button will pause the play of a sound message. The button acts as a toggle so that play can be resumed by pressing the button again. The pause button will only be enabled during playing of the message.

2.5.6 Play Button

The play button will play the message. The button will be enabled after the message has been recorded or if the user in retrieving a saved message.

2.5.7 Play/Record Progress Indicator

The play/record progress indicator will display the time elapsed as a progress bar during a play session. The max. number will indicate the total time length of the sound byte. During a record session the bar will indicate the ratio of disk space consumed over the space available. The max. number will indicate the total space available on the disk which holds the application.

2.5.8 Play/Record Time Elapsed

The message time elapsed during a play or record session will be displayed in the play/record time elapsed field.

2.5.9 Record Button

To record a new message and save it to the current image press the record button.

2.5.10 Record Space Consumed

The record space consumed will display a running total of the disk space consumed during a record session.

2.5.11 Rewind Button

Rewind the voice play.

2.5.12 Save Button

To release the dialog and save the sound message to the image file press the Save button.

2.5.13 Sound Level Indicator

The relative sound level will be illustrated in the sound level indicator via a series of 5 "Sound Wave Lines". This gives the user an indication of the amplitude of the sound wave.

The Five Sound Wave Lines

2.5.14 Speed Control

The speed control sets the playback rate. The control is initially set to 1 (at the center point of the control) for normal playback. The control has a speed increase range from normal (1x) playback to 8 times normal playback in 12 increments with an exponential base of 2 (ie. from $2^{\wedge 0}$ to $2^{\wedge(12/4)}$). The speed decrease range is equal to the inverse of the increase range (ie. from 1 to 1/8th speed). The default setting is 1.

2.5.15 Stop Button

To stop the recording or playing of the message press the stop button.

2.5.16 Tone Control

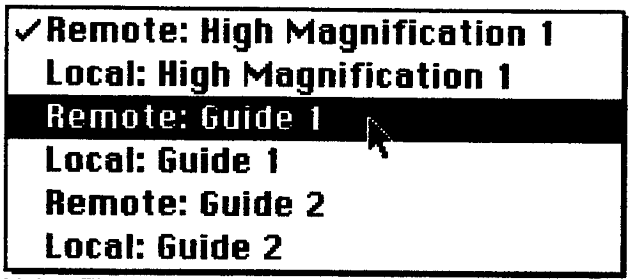
The tone control allows the user to set the tone of the sound. This feature allows the user to adjust the sound pitch so that the voices can be understood at different speeds. The tone is

measured by the amplitude of the sound wave. The scale and range of the control is identical to the speed control. The default setting is 1.

2.5.17 Transcription Field

During transcription this field will be available and capable of receiving keyboard input. As this is a standard Macintosh editing field it will support all of the standard edit features such as Cut, Copy, Paste, and Clear.

2.5.18 Voice File Popup



Voice File Popup Menu

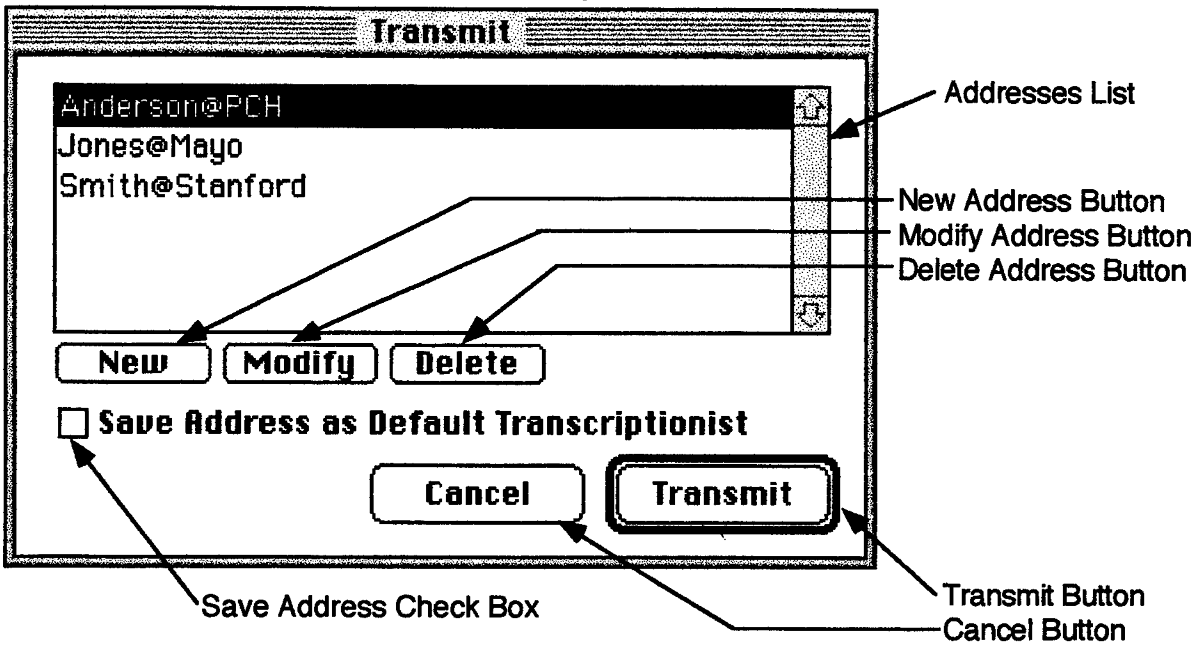
To select a different file for recording or playing the user will choose an item from the voice file popup. This popup allows the user to play the general directions left by the other user (which is associated with the guide image) no matter what the current high magnification location is. The user can also record over or append to the general directions from any high magnification location. The voice file chosen when the user invokes the dialog will be based upon an educated guess as to what the user wants. For example, if the user is at a new location during high magnification creation and has not yet played the voice from the other user then the dialog would naturally choose the "Remote: High Magnification 1" file. However, if the user had already played the voice then the file chosen would be "Local: High Magnification 1". Note that this Local file does not exist yet and if the user does not record voice at the current location then a voice file never gets created, the popup item is simply there to allow the user to create the voice. If the user wants to listen to any voice from previous iterations (should they exist) they would be available in this popup. Of coarse the user can only record to their own station (local or remote) and only to their current iteration. At all other times the files are read only.

2.5.19 Volume Control

The volume control has a range from 0 to 100 in increments of 1. A value of zero represents no sound and 100 represents the current setting of the Macintosh sound control panel. The default setting is 100.

2.6 Transmit Dialog

To send an image request or transmission to another site the user will invoke the transmit dialog. This dialog is a moveable modal dialog.



Transmit Dialog

2.6.1 Addresses List

The address list will display all of the saved remote site addresses. These addresses will be stored in a flat file for easy transportation and modification.

2.6.2 Delete Address Button

To delete the selected address press the delete address button.

2.6.3 Modify Address Button

To add a new blank address or to save the current ad-hoc address press the New Address button.

2.6.4 New Address Button

To add a new blank address or to save the current ad-hoc address press the New Address button.

2.6.5 Save Address Check Box

This check box is only available during a transcriptionist selection session. Otherwise, the check box is hidden. If the user selects the check box then the address will be saved to the preferences file as the current users default transcriptionist.

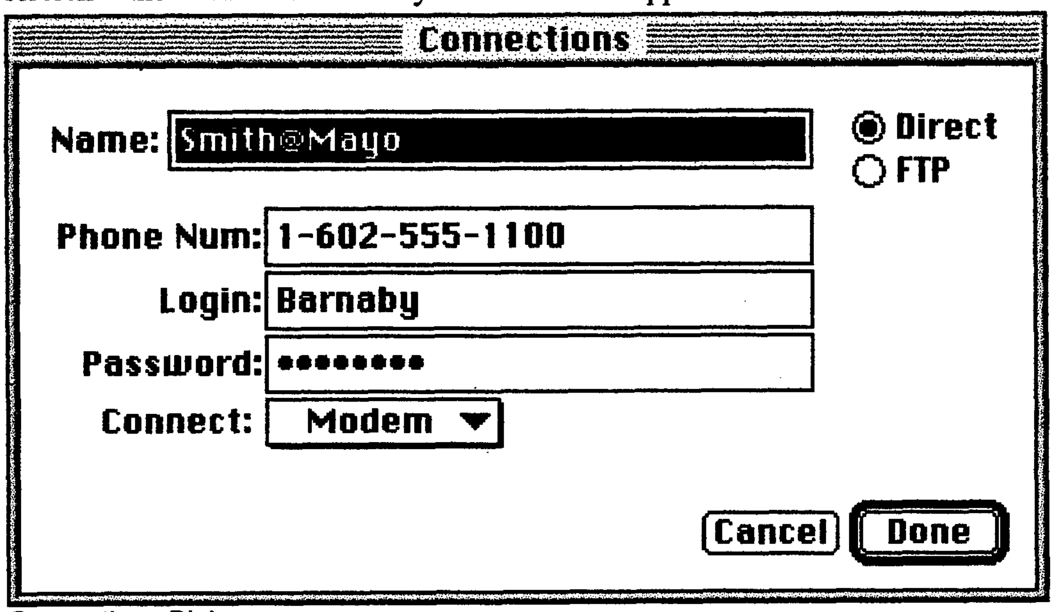
2.6.6 Transmit Button

After selecting the recipient press the transmit button to initiate the transmission object.

2.7 Connections Dialog

The connections dialog will allow the user to name connections and addresses to various remote users. The dialog will be capable of handling many types of communication protocol. The fields and buttons will change according to the protocol chosen so no discussion is given at this time. At a minimum the dialog will be configured to support point to point communication over ISDN. The constants in the dialog are described below.

The communication object (TPWCommunications) will be responsible for handling all communication functions. This includes the sending and receiving of files as well as the transmission of mouse and keyboard activity during chat. All of the functions inside of the communication object will be pure virtual meaning that they must be overridden. This is to ensure that the functionality is provided regardless of the machine or communication type. Currently, the only proposed connection type is ISDN over phone line. Since the development is being done on the Macintosh it is planned that the communications be maintained using Open Transport. Open Transport is currently a set of INIT's which will eventually become a standard part of the Mac OS. Open Transport provides all of the communication functionality which will be required by the TPW and continues to add connectibility for other connection types. For more information on Open Transport reference the documents currently available from Apple.



Connections Dialog

2.7.1 Cancel Button

Discard the changes to the connection.

2.7.2 Done Button

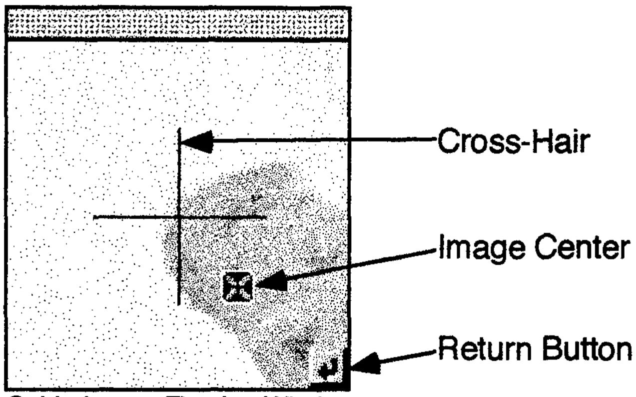
Accept the changes to the connection.

2.7.3 Name Field

Enter a meaningful name for the recipient.

2.8 Guide Image Floating Window

The guide image floating window is only available during high magnification creation. The window displays an image 256x256 pixels. The cross will always indicate the center of the current high magnification location so that the user can easily navigate.



Guide Image Floating Window

2.8.1 Cross-Hair

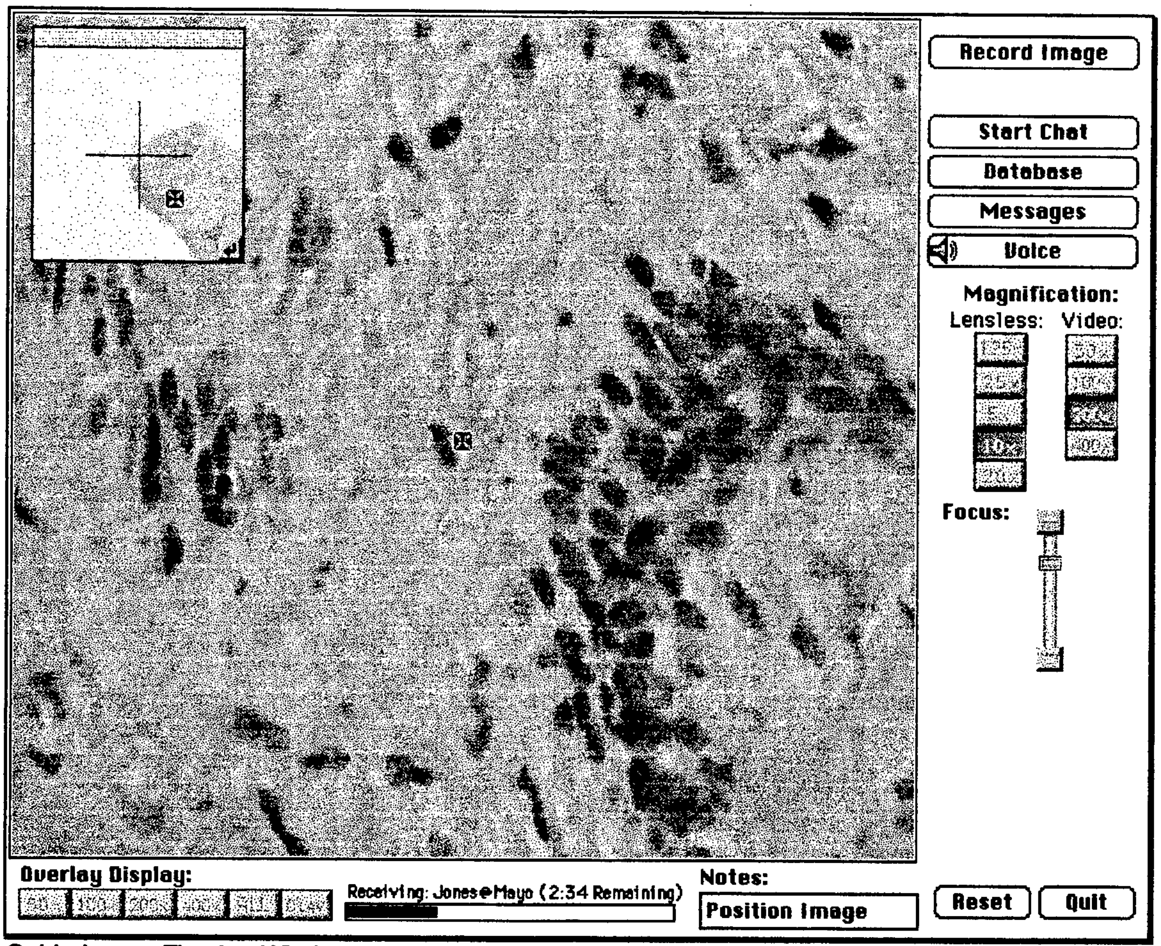
The cross-hair is always in the center of the floating window which always displays the image at the center of the remoter users request. The cross-hair indicates the center of the requested location from the remote user.

2.8.2 Image Center

The image center indicates the center of the users current location in the main window high magnification image display.

2.8.3 Return Button

The return button will reposition the user at the location of the requested high magnification image. If the user is at the exact location of the request the button will be disabled.



Guide Image Floating Window In Use

2.9 Database Entry Dialog

The database entry dialog will be available after all dictated diagnoses have been transcribed and before the diagnosis information is saved to disk.

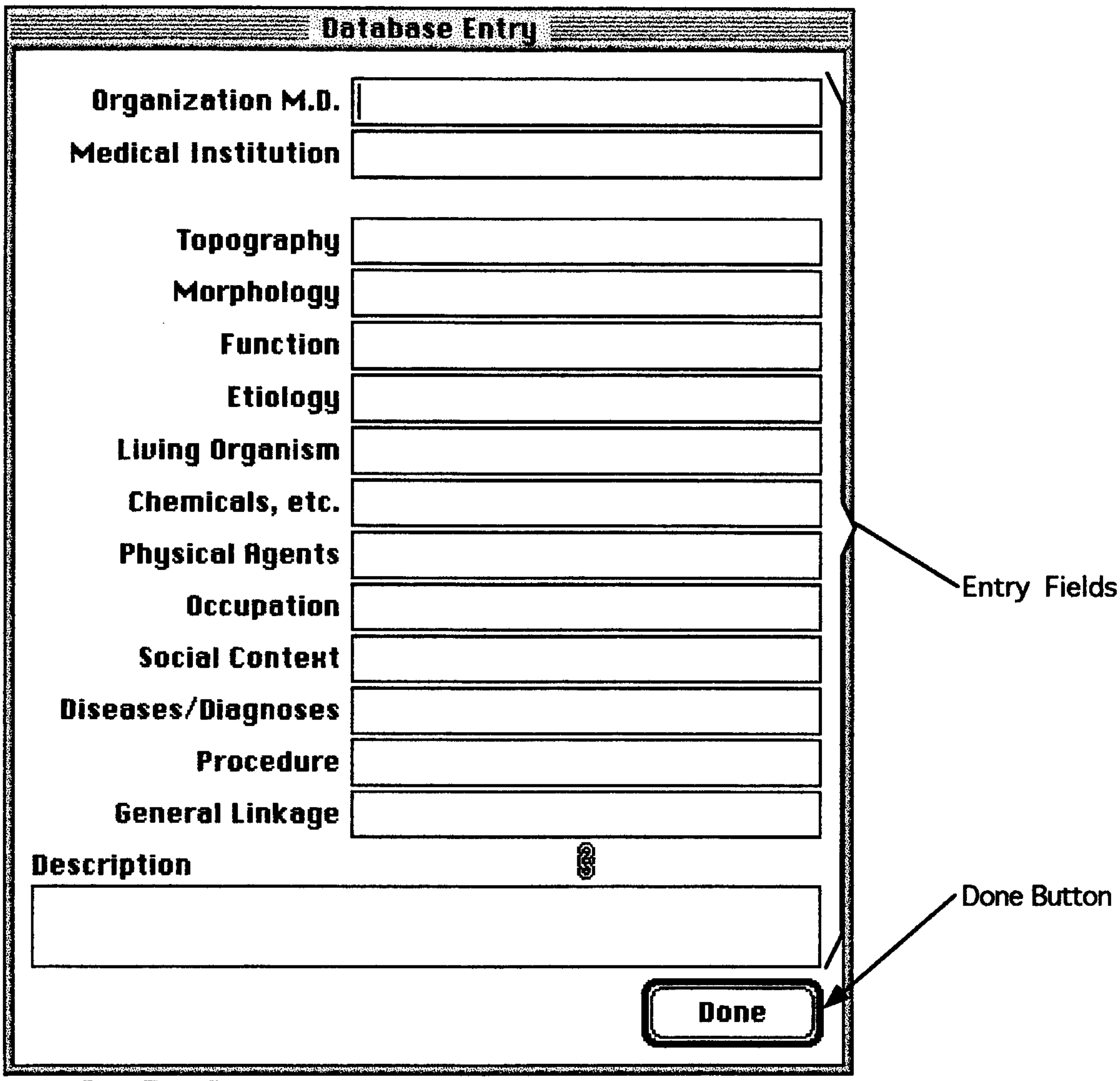


Image Data Entry Dialog

2.9.1 Done Button

The only button available is the done button. This is the only screen which allows the user to enter this information and the user only has one chance at it. It is therefore not desirable to have a cancel button for the user to avoid entry.

2.9.2 Entry Fields

All data will be entered through the entry fields. The MD. and Institution fields are simply entry fields. The Topography through Occupation fields (SNOMED generators) are actually tied to the SNOMED Code field. When a value is entered into one of the SNOMED generators the appropriate entry into the SNOMED code field is automatically made. Likewise, when the SNOMED code field is edited the corresponding generator field will be automatically updated.

2.10 Image Data Floating Window

The image data floating window is only available during database mode, and only when the main window is the top window. The window displays the image data for the selected image. The window is not closeable because their is no mechanism for re-opening it. It can always be moved out of the image view and is short enough to fit under the view unobtrusively.

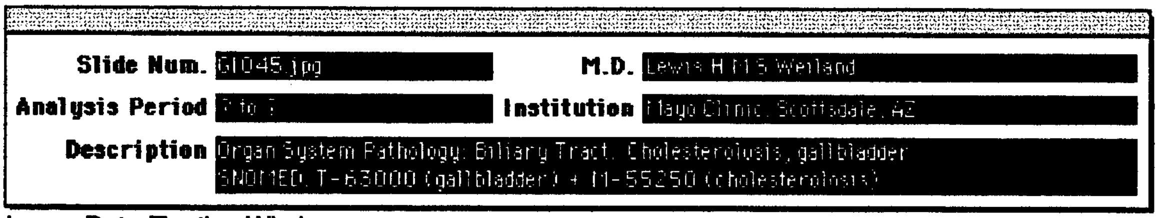
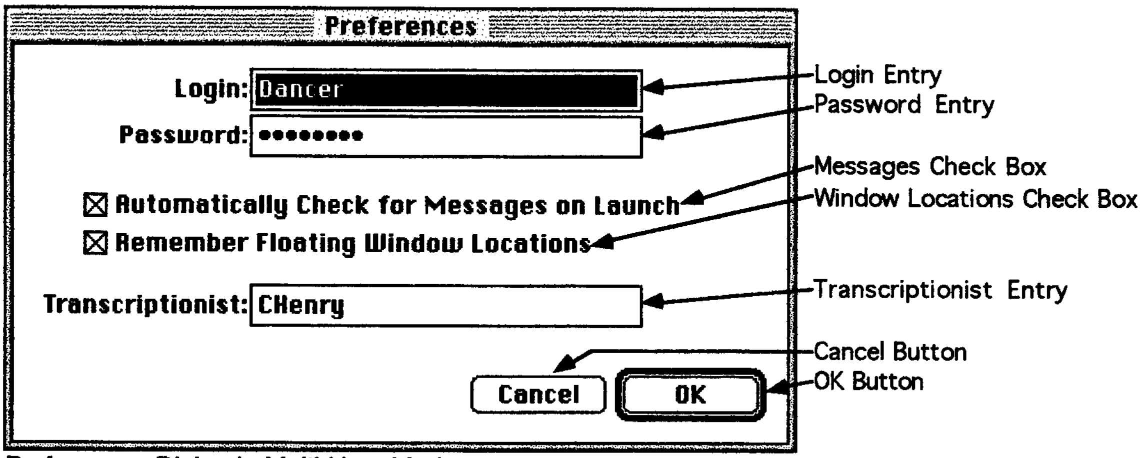


Image Data Floating Window

2.11 Preferences Dialog

The preferences dialog is accessed by opening the preferences file which will be in the same folder as the application. The user can open the file whether the application is currently running or not. If the application is not currently running then it will be launched. Currently the only preferences which have been identified is the ability to change or delete the saved password. As other preferences are identified during the development process they will be incorporated into the dialog.



Preferences Dialog in Multi User Mode

2.11.1 Cancel Button

To discard all changes to the preferences dialog during the current session the user presses the cancel button.

2.11.2 Login Entry

If the system is in single user mode the user can change their login name in the login entry.

2.11.3 Messages Check Box

The user sets the messages check box on in order to have the system automatically prompt for message response on system launch.

2.11.4 OK Button

To save all changes to the preferences dialog during the current session the user presses the OK button.

2.11.5 Password Entry

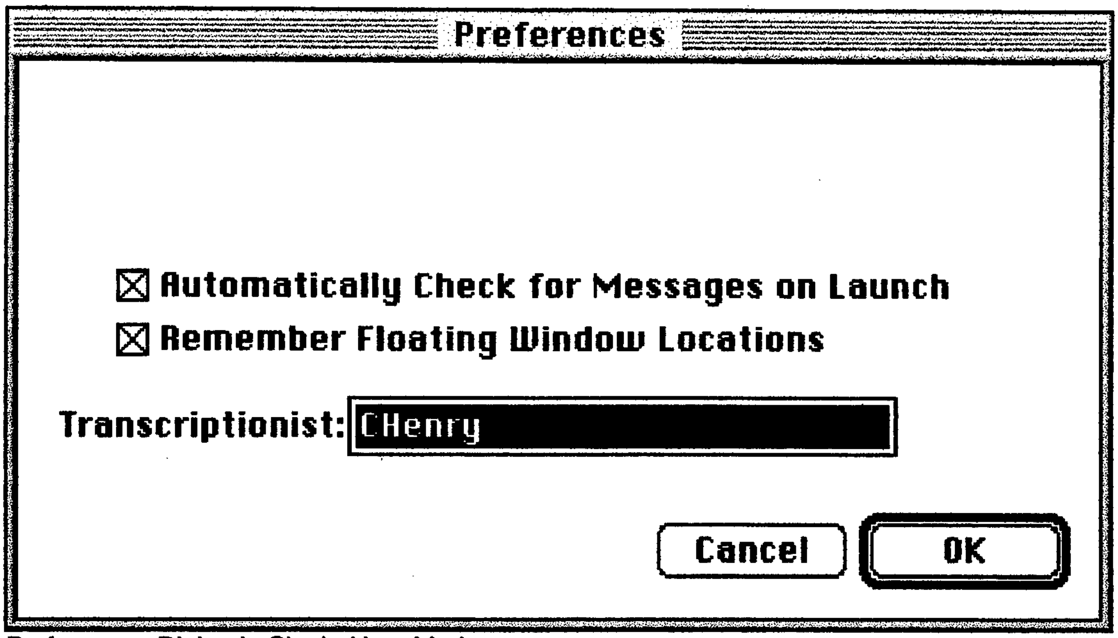
If the system is in single user mode the user can change their password in the login entry.

2.11.6 Transcriptionist Entry

The user can select a new transcriptionist or delete the current one (thus requiring a selection each time the diagnosis is sent) in the transcriptionist entry field.

2.11.7 Window Locations Check Box

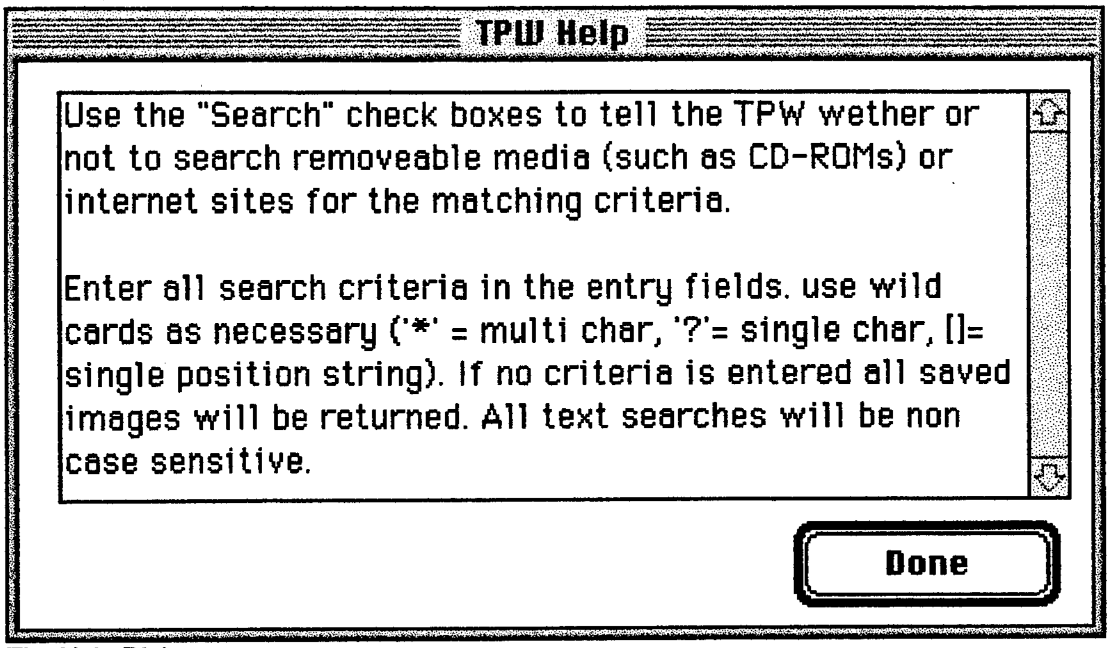
In order for the floating window locations to be saved from session to session the user will check the window locations check box.



Preferences Dialog in Single User Mode

2.11 Help Dialog

The help dialog is currently invoked only from the database search dialog.

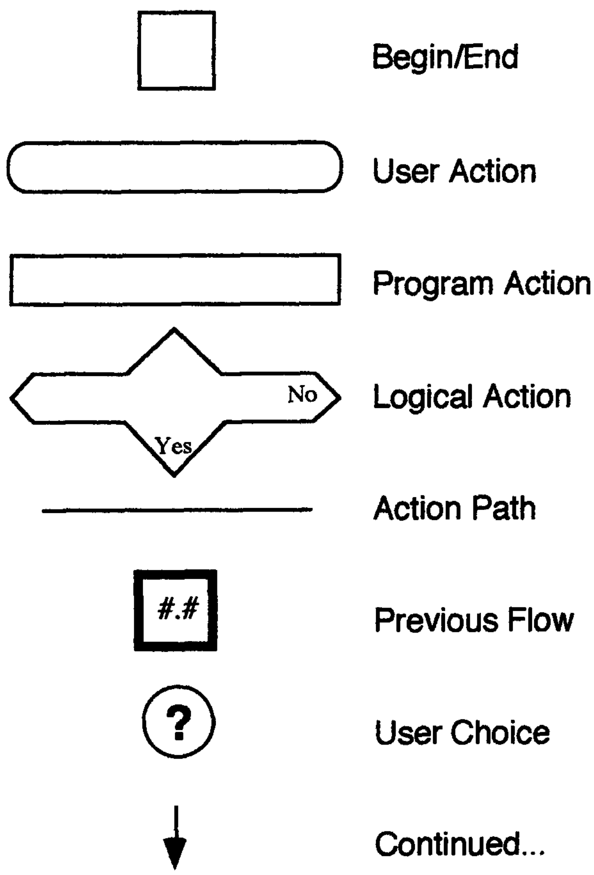


The Help Dialog

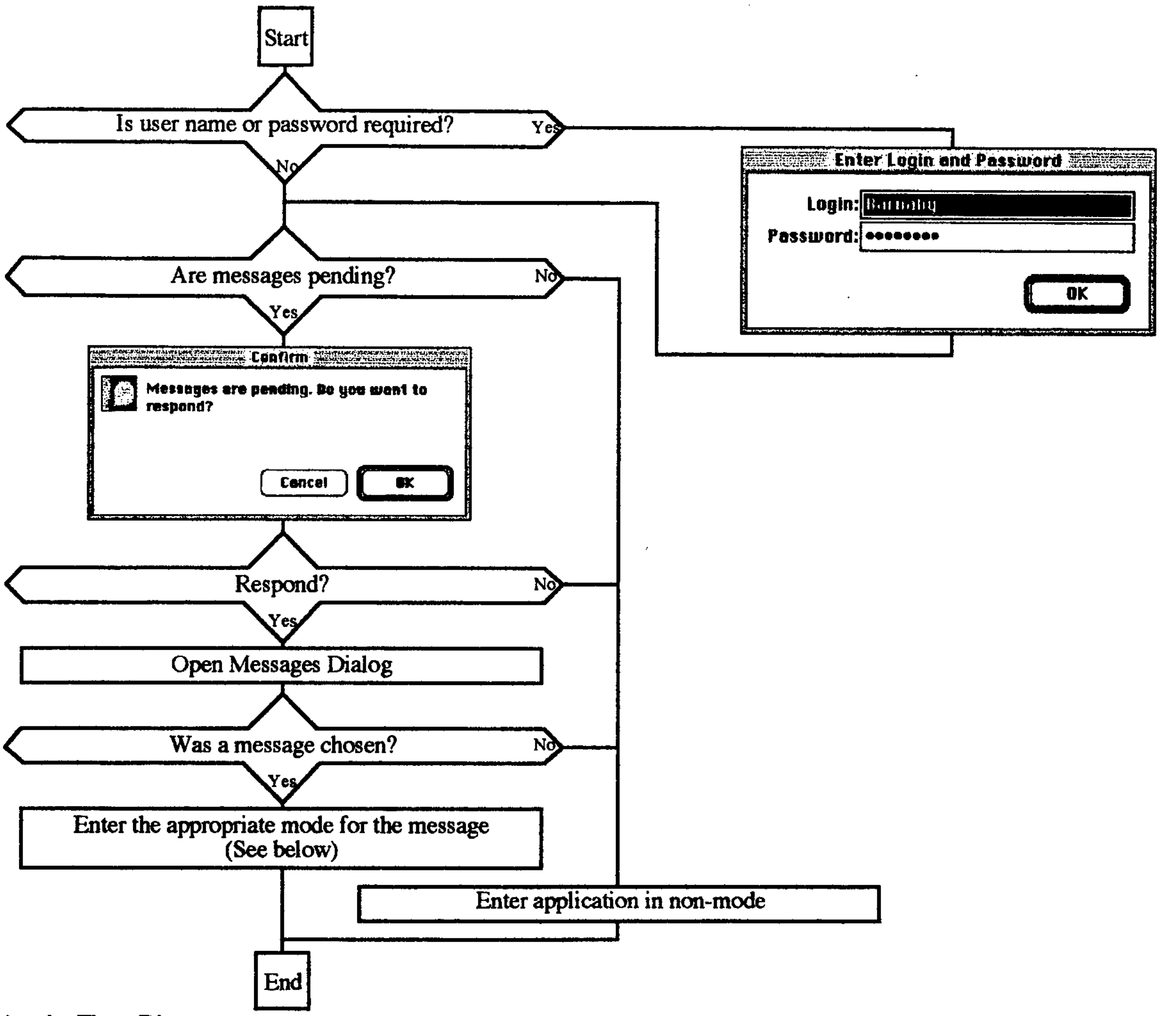
3 USER SCENARIOS

The following scenarios outline the common Tele-Pathology user scenarios from a graphical and procedural stand point. For the purpose of this discussion the Local User will be known as the user who possesses the physical specimen slide in a dual station scenario, and the Remote User is the person who has access to the slide only through the generated images.

Legend

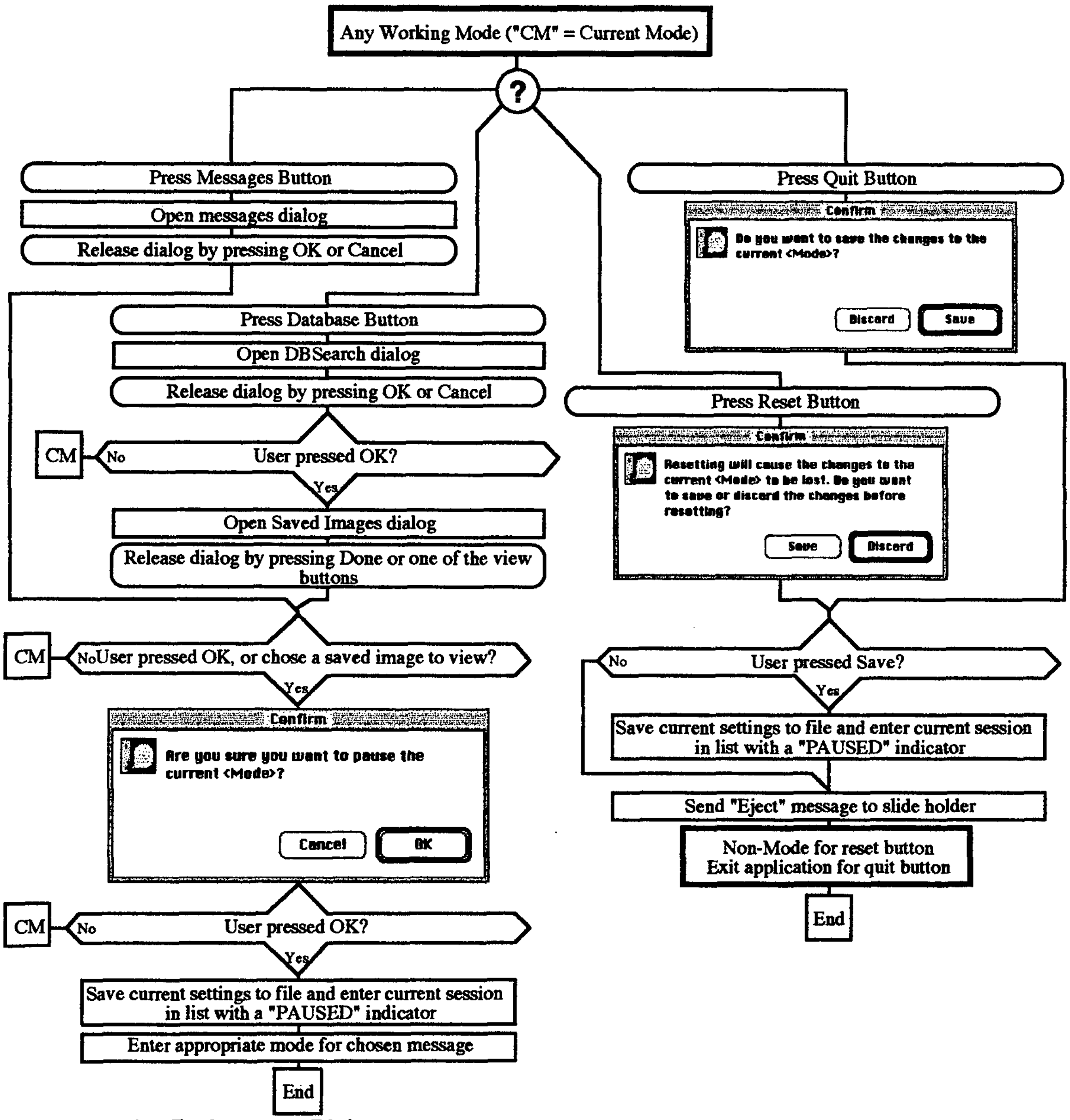


The following scenario depicts the login events. In this scenario the user has launched the application and will be prompted for appropriate actions.



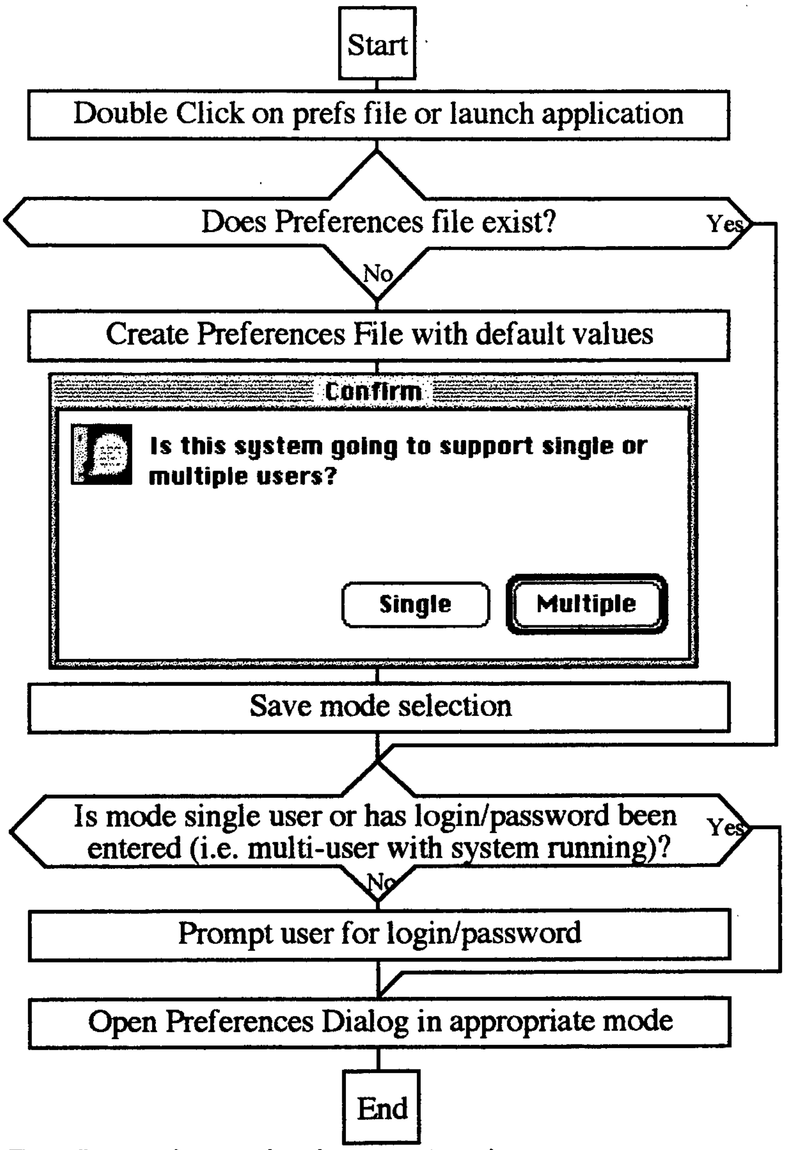
Login Flow Diagram

Another potential startup scenario is that of double clicking on the preferences file or launching the application when the preferences file does not exist (or cannot be found). Either of these two cases can be used to get to the preferences dialog and save preferences. By looking at the following diagram the reader can quickly identify that the only way in which a system can be changed between single and multi user is to throw away the preferences file (or if the user is familiar with the preferences file format, by editing it directly).



Accessing the Preferences Dialog

During any of the working modes the user could choose an action which will require either saving and pausing the current scenario or discarding the changes. The working modes consist of: Create Guide, Request High Magnification Images, Create High Magnification Images, Generate Diagnosis, and Slide Examination. The actions which may initiate this flow are: pressing the messages button (if messages are available), pressing the database button, pressing the reset button, pressing the record button if the user is in create guide mode, or pressing the quit button. If the user chooses one of these actions then they will be prompted for a decision to put the current mode on pause.



Flow diagram for pausing the current mode

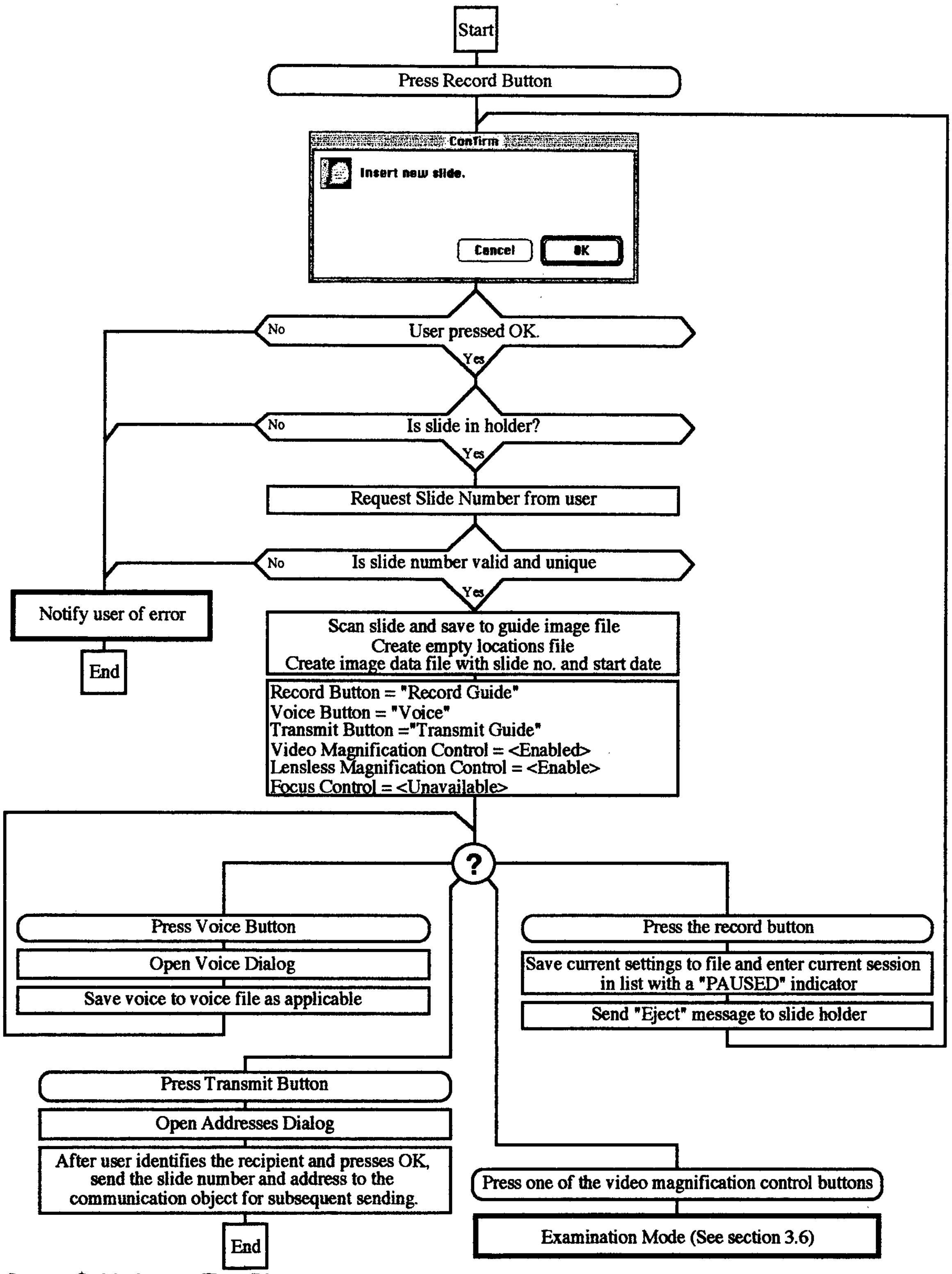
The user can also pause the current mode by entering a chat (see section 3.8) or by quitting the application.

All following scenarios assume that the application is currently running. In each of the scenarios the indication of the "End" means that the user enters non-mode. In this mode the control states are as follows:

Record Button	"Record Guide"
Voice Button	<Unavailable>
Transmit Button	<Unavailable>
Video Magnification Control	<Unavailable>
Guide Magnification Control	<Unavailable>
Focus Control	<Unavailable>

3.1 Dual Station, Local User - Generate Guide Image and Transmit

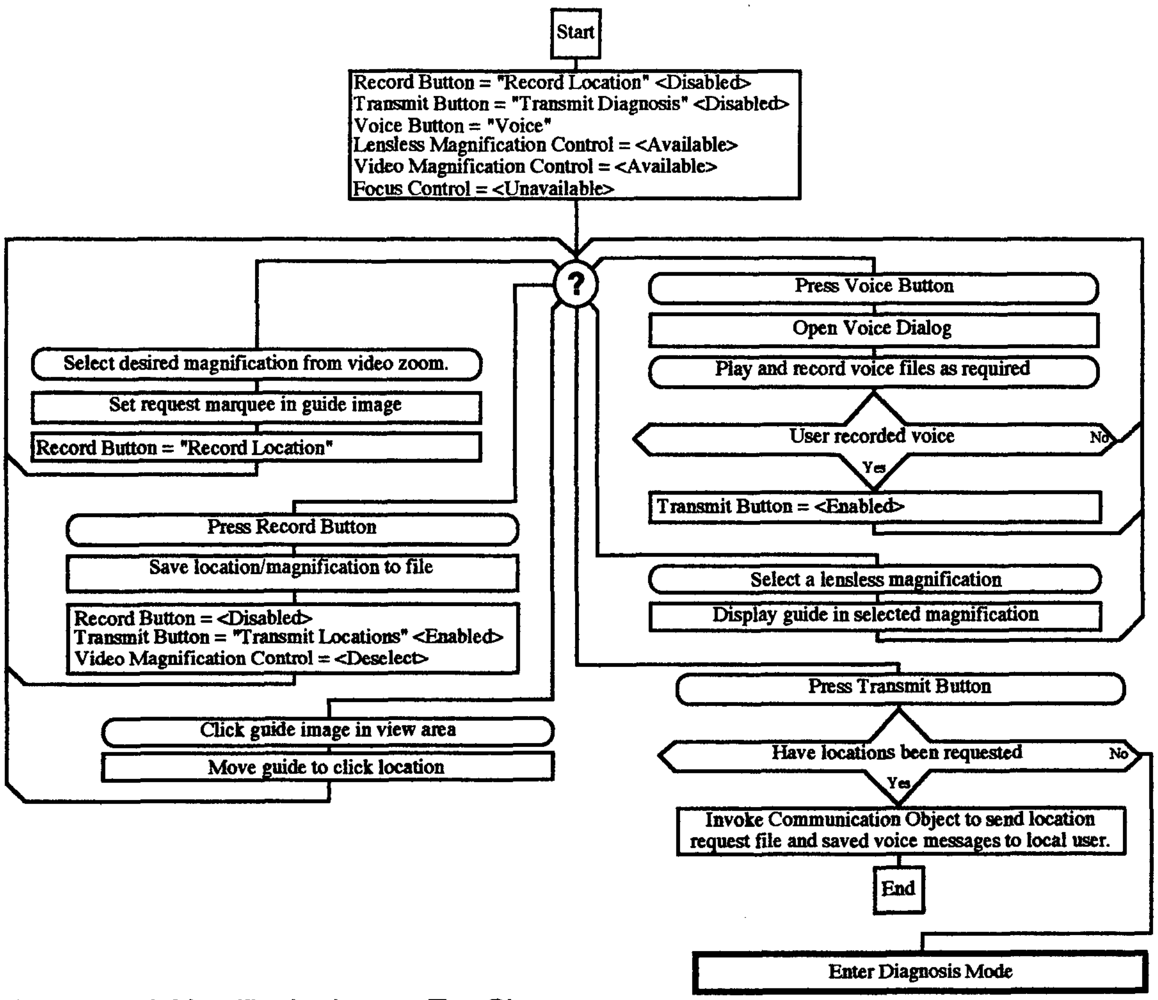
The local user begins by loading the slide onto the table and pressing the record button which reads "Record Guide". The system then prompts the user for the slide number and scans the slide at 1x magnification. If a slide is not on the table or an invalid slide number is entered the system displays an error and no Mode is activated. At any time the user could press the voice button to store a voice message with the image. After recording the image and optional message the user presses the transmit button and chooses a recipient from the transmit dialog. The session is concluded with the successful transmission of the image to the remote user.



Create Guide Image Flow Diagram

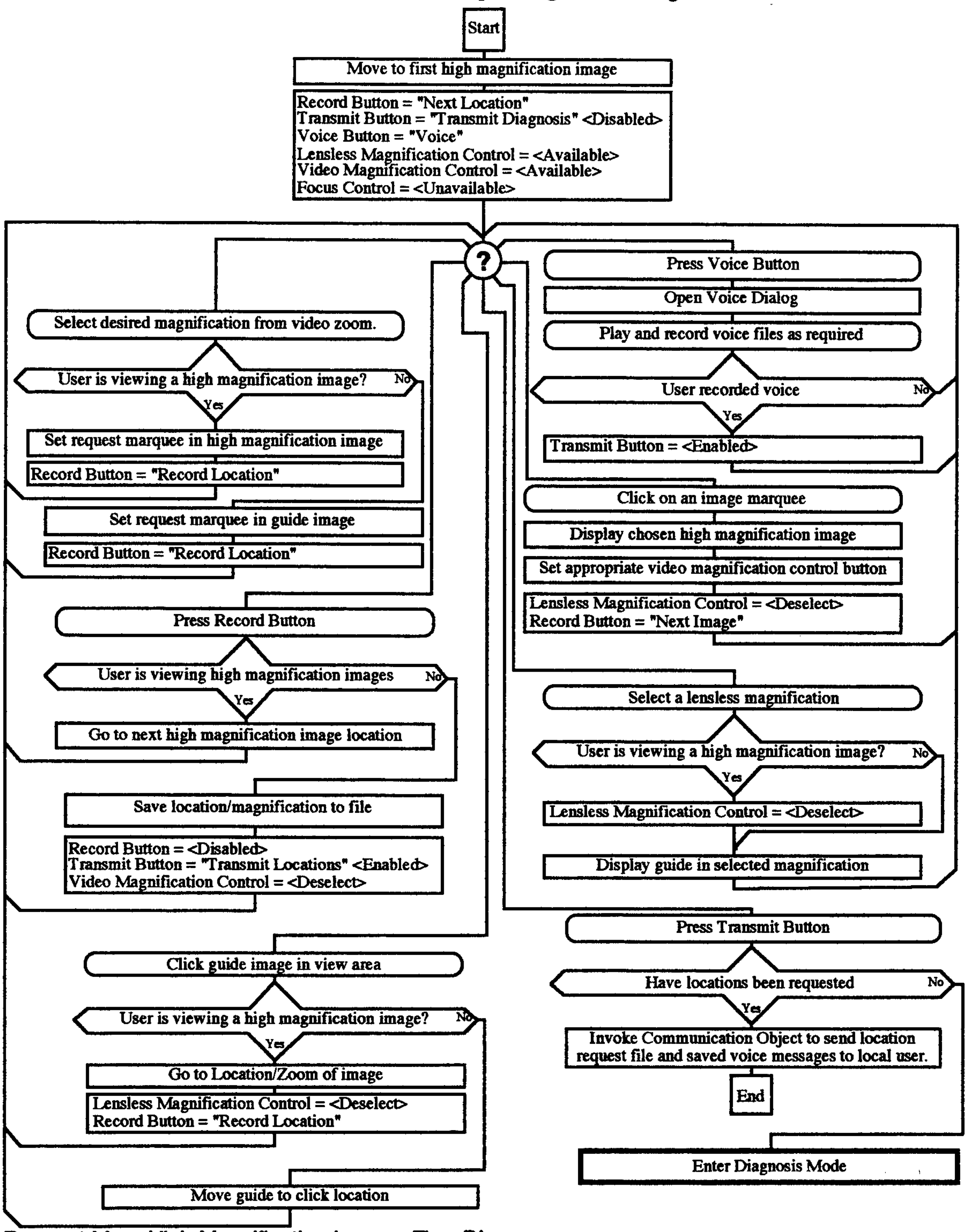
3.2 Dual Station, Remote User - Request High Magnification Images

The remote user will notice a message in the message list indicating that a guide image has been received and analysis requested. The user can either double click on the message or press the respond button. The user will now be in the request high magnification images mode. The user will review the image on screen and choose locations for high magnification analysis by pressing the record button (titled "Record Location"). Before pressing the record button the user will have positioned the area of interest in the center of the view and chosen an appropriate magnification with the video magnification control. The view will display a floating dotted rectangle (marquee) indicating the area of the image which will be captured by the camera when the high magnification image is created. The user can select a magnification from the guide magnification control at any time to see more or less of the guide image. The marquee will adjust according to the users position and magnification. Once the user presses the record button the dotted rectangle will fix to the guide and be added to the locations list. The video magnification control will un-depress and the floating rectangle will disappear until the user chooses another high magnification setting. The user can review any requested location/magnifications by double-clicking on its marquee number in the guide display. When a location is selected for review the dotted line and magnification will be set and the record button will read "Remove Location". As soon as the user scrolls the image the record button will again read "Record Location" and the video magnification control will be un-set. The record button will only be activated if the video magnification is set to something. At any time the user can attach a voice message to a location by choosing the location and pressing the voice button. If a location has already been associated with a voice message the user will be able to play the message or record over it. If the user creates a message at any time that the video magnification is un-set this message will be attached to the session in general as opposed to attaching to a specific location. Each requested locations center point within the guide image and magnification will be stored in a flat file. After all locations of interest have been identified the user will transmit the request file and messages to the local user by pressing the transmit button. This action will always return the file to the same local user which sent the guide image.



Request High Magnification Images Flow Diagram

The following diagram illustrates the scenario when the user has requested images, received them, reviewed them and is now requesting more images.



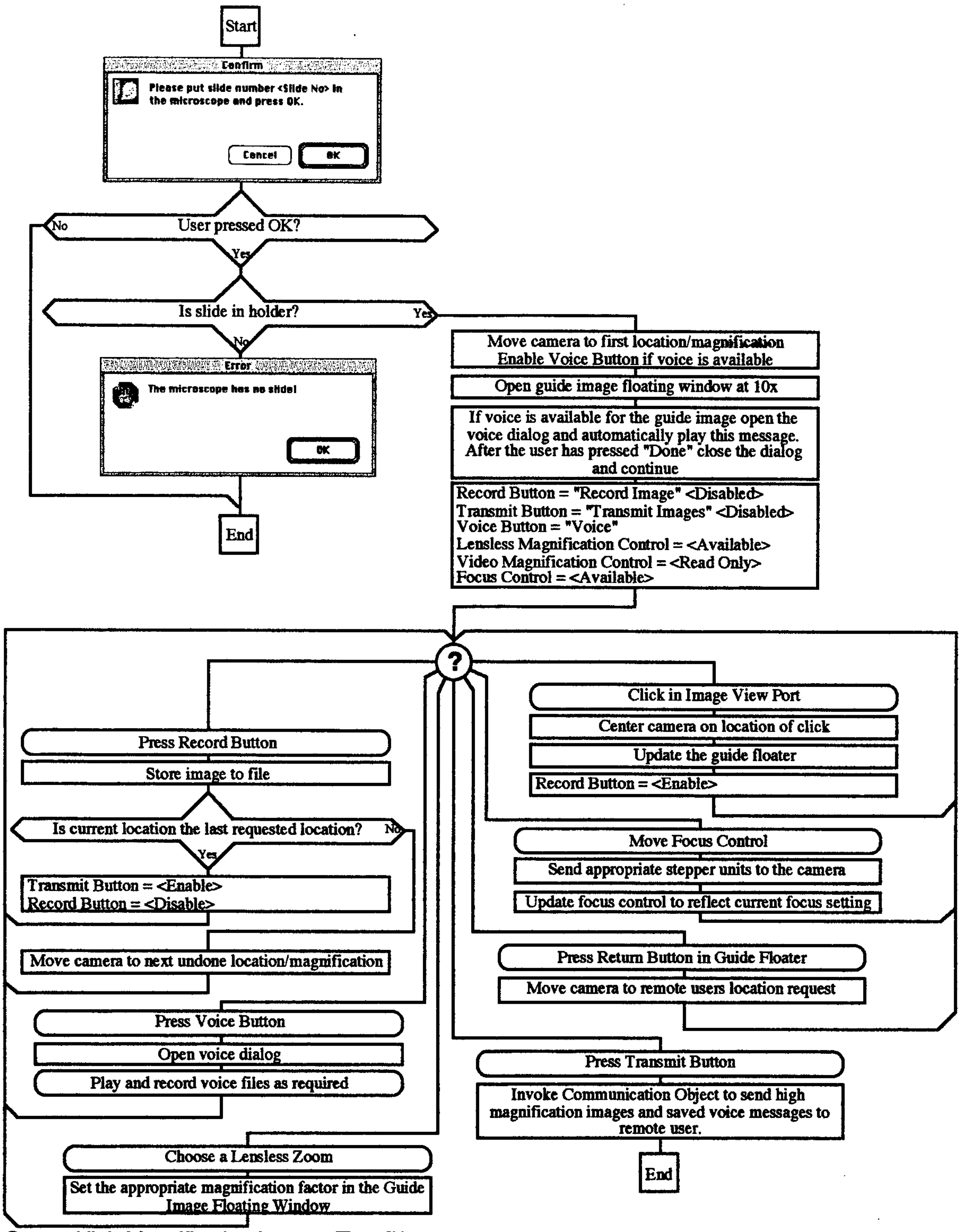
Request More High Magnification Images Flow Diagram

3.3 Dual Station, Local User - Generate High Magnification Images

After the message has been received by the local user the message list will display a notification that high magnification images have been requested. To respond to this message the local user will select the item and press the respond button. The system will then request the slide number and after successfully retrieving the guide image, checking for a valid slide number and ensuring that the slide is on the table will take the user to the "Create High Magnification Images" mode. Each of the location requests will be indicated by a cross which will disappear after the user creates the image. During high magnification image creation the user will have a floating window which will display the current location request at 10x initially. The view window will contain the image at the requested magnification.

The user will automatically be taken to the first location/magnification request and be allowed to pan and scroll and set the focus before pressing the record button. At each location the camera will be automatically magnified and the position scrolled according to the request. The focus will be automatically set to maximize contrast. The user can listen to any messages from the remote user by pressing the voice play button. Before pressing the record button the user can save a message for the remote user by pressing the voice record button.

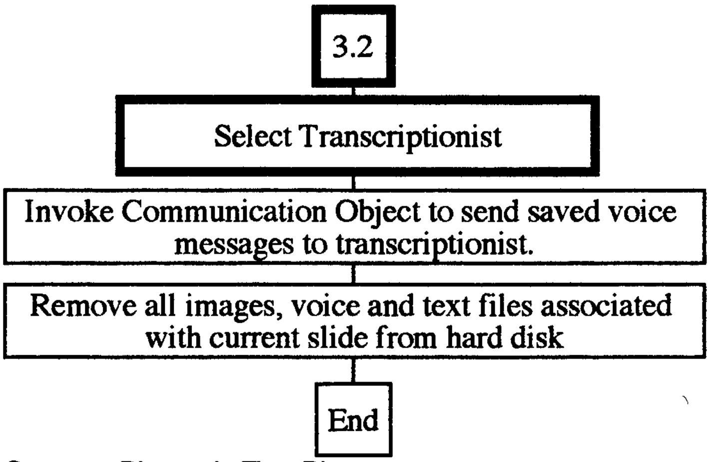
After pressing the record button the user will be taken automatically to the next location. The cross will also dim to signify to the user that the location has been completed. During the entire scenario the transmit button will be disabled until the user records the final image. After pressing the transmit button the images and messages will be sent to the remote user who requested the images.



Create High Magnification Images Flow Diagram

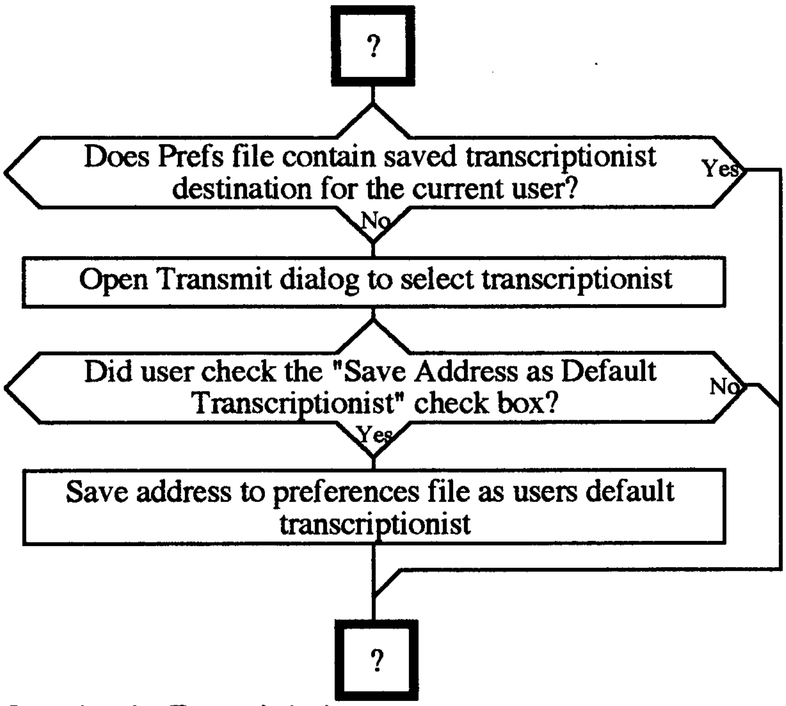
3.4 Dual Station, Remote User - Diagnose From Images

The messages list will notify that the high magnification images have been received from the local user and are ready for analysis. The record button will allow the user to step through the images by acting as a next button (record button is titled "Next Image"). The user can also move from location to location by clicking on the marquee numbers. The user can listen to messages left by the local user by pressing the voice play button. During the viewing of high magnification images the voice record button will invoke the voice dialog for the remote user to dictate a diagnosis associated with the current image. The user moves to the guide by selecting a magnification from the guide magnification. The diagnosis dictation will be sent for transcription to the transcriptionist when the user presses the Transmit button.



Generate Diagnosis Flow Diagram

The following flow diagram illustrates the procedure in selecting the transcriptionist

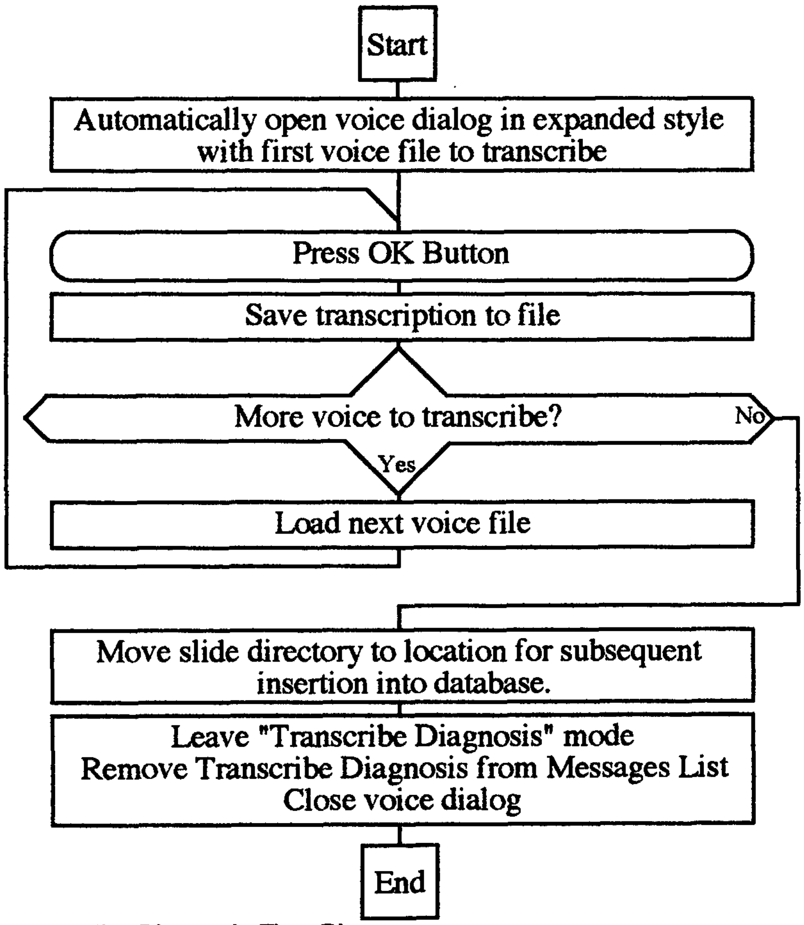


Selecting the Transcriptionist

3.5 Dual Station, Local User - Transcribe Diagnosis

After receiving the dictated diagnosis from the remote user the local user will review the messages and have them transcribed into a text file to be stored with the images. The transcribed voice file will contains section headers indicating the location/magnifications where the diagnosis was made. These headers will be automatically inserted via the transcription process.

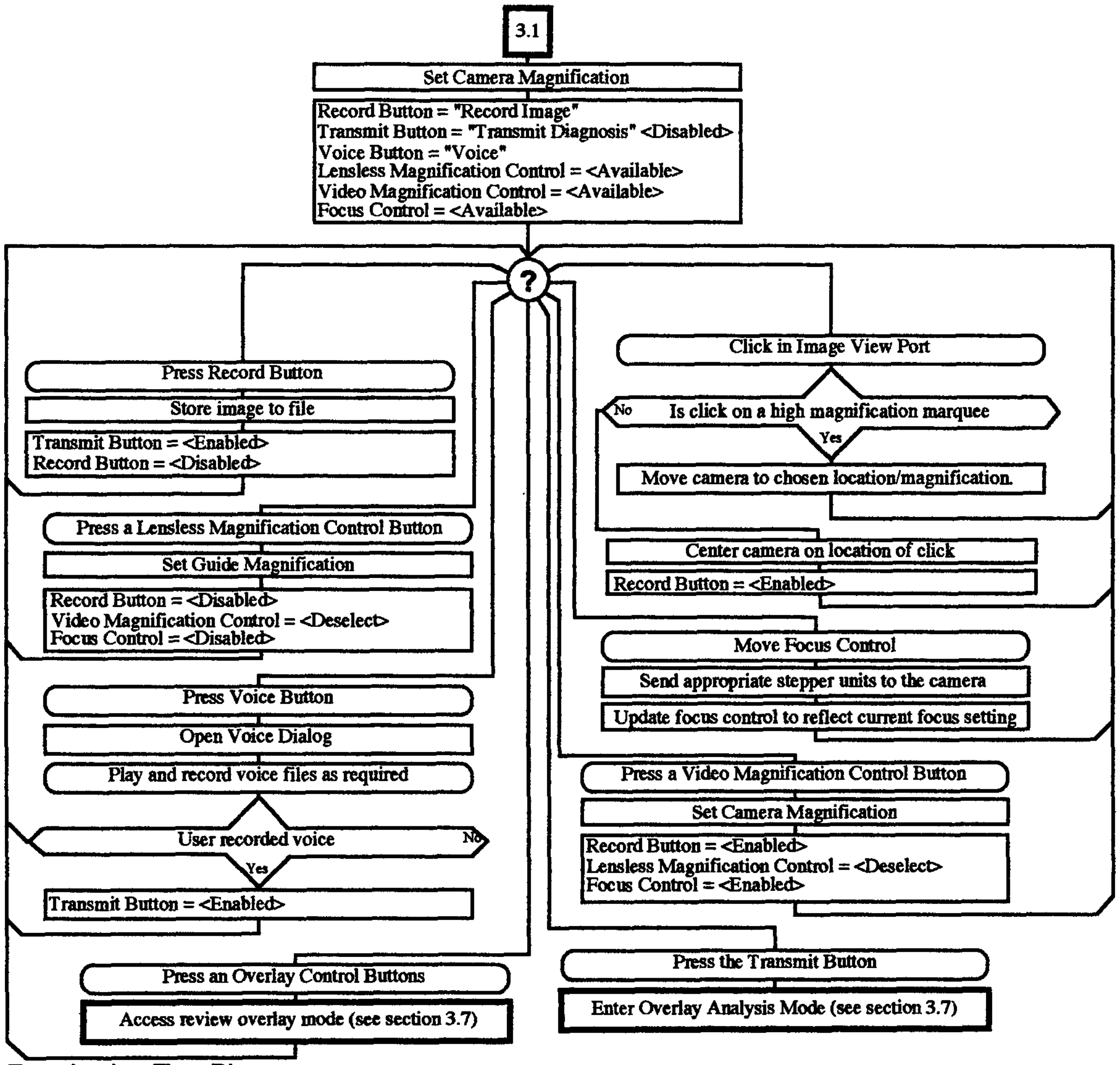
To transcribe the message the local user will select the transcription request from the message list and press the respond button. The user will be automatically taken to the voice dialog in expanded mode with the first diagnosis. When the voice dialog is expanded the text entry will always be live so that the user can press play, pause or stop without losing the current position in the text. During play of the message the text entry will be live so that the local user can type and listen at the same time. After the current dictation is transcribed the user presses the save button (reading "Next") and is automatically taken to the next dictation. After having transcribed all messages and pressing the save button the system will prompt the user for database information and automatically terminate the session by saving the diagnosis and image data to file.



Transcribe Diagnosis Flow Diagram

3.6 Single Station - Examination

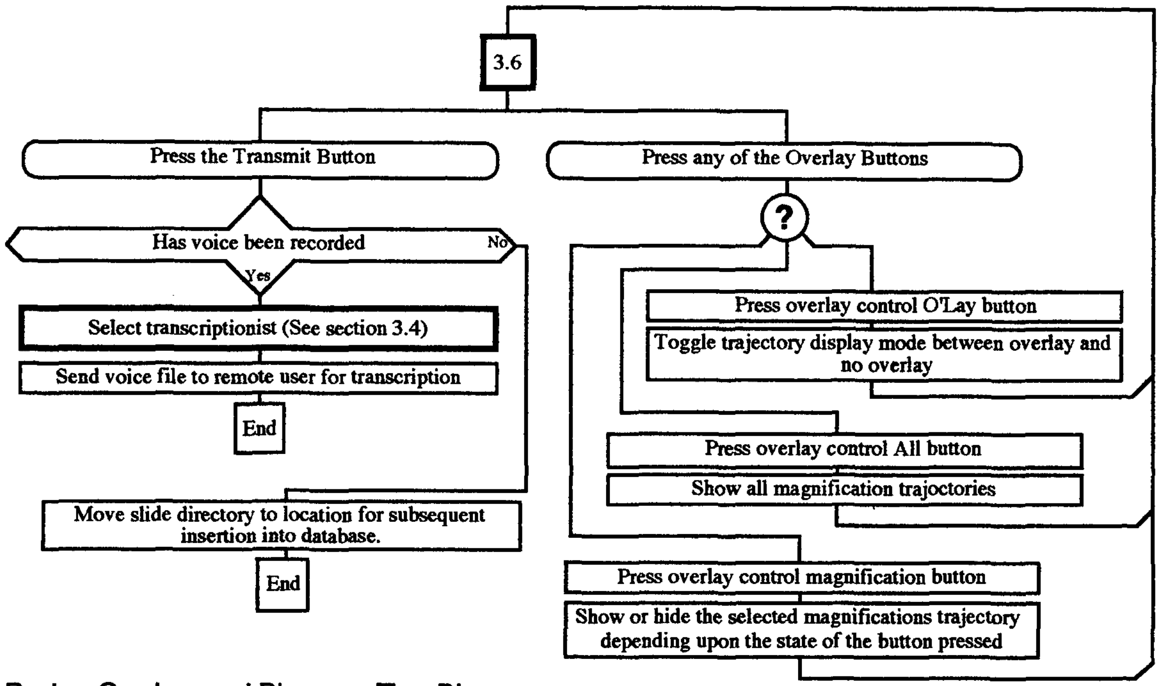
In the final two creation scenarios the local user is also the expert so transmission is not necessary. The user reviews the slide at any location and magnification desired and records images and/or voice at chosen locations. The overlay is stored in a high magnification image request file which logs the location and the magnification of the regions of interest. The locations file stores each and every location/magnification which the user traverses. The file makes note of any locations at which the user records a high magnification image. The user reviews their overlay by pressing one or more of the overlay buttons.



Examination Flow Diagram

3.7 Single Station - Review Overlays and Diagnose

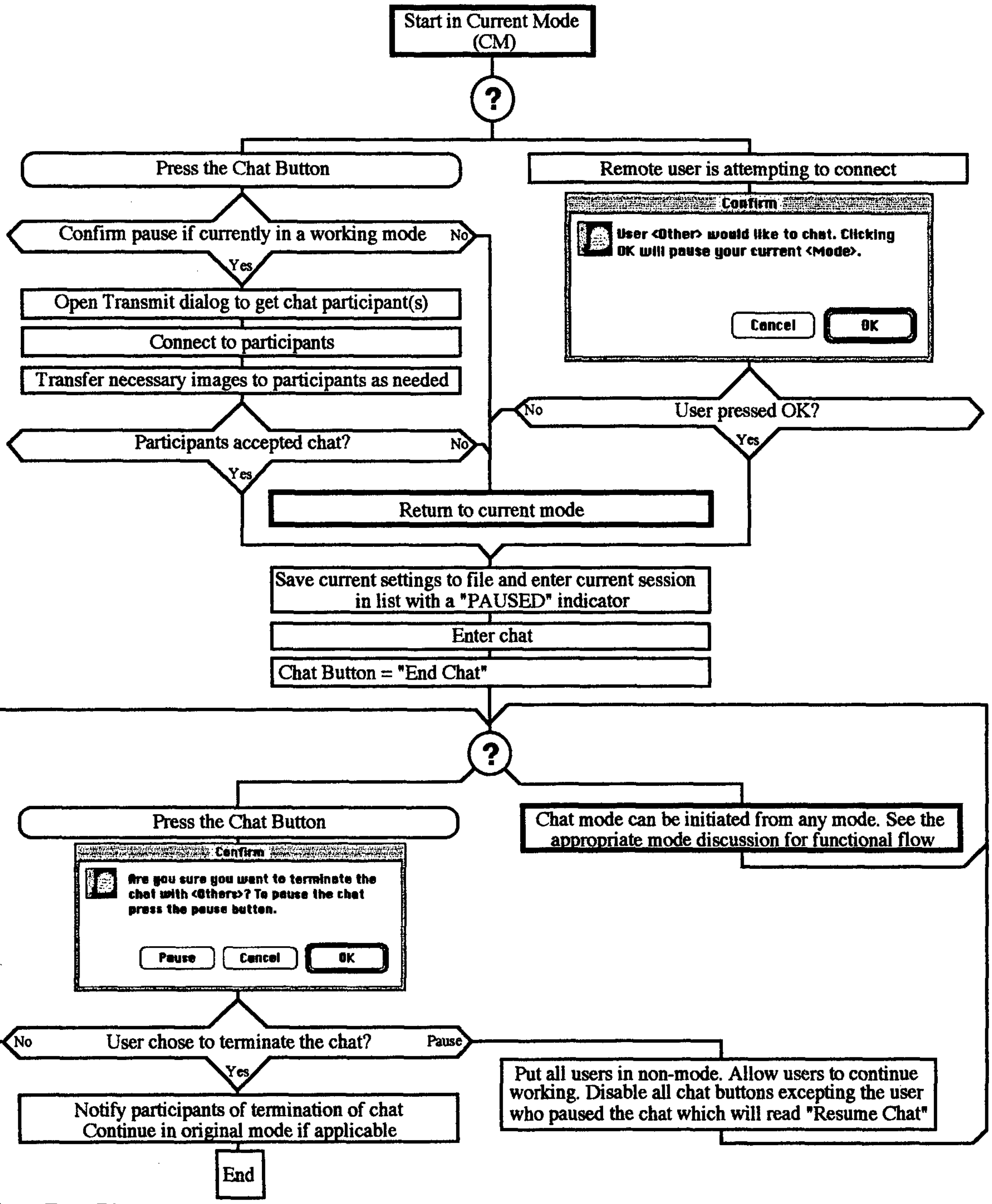
After having traversed the slide as needed the user can review the overlays and dictate a final diagnosis. This diagnosis will then be transcribed and stored along with the images at each location under scenario 3.5.



Review Overlays and Diagnose Flow Diagram

3.8 Dual Station - Chat

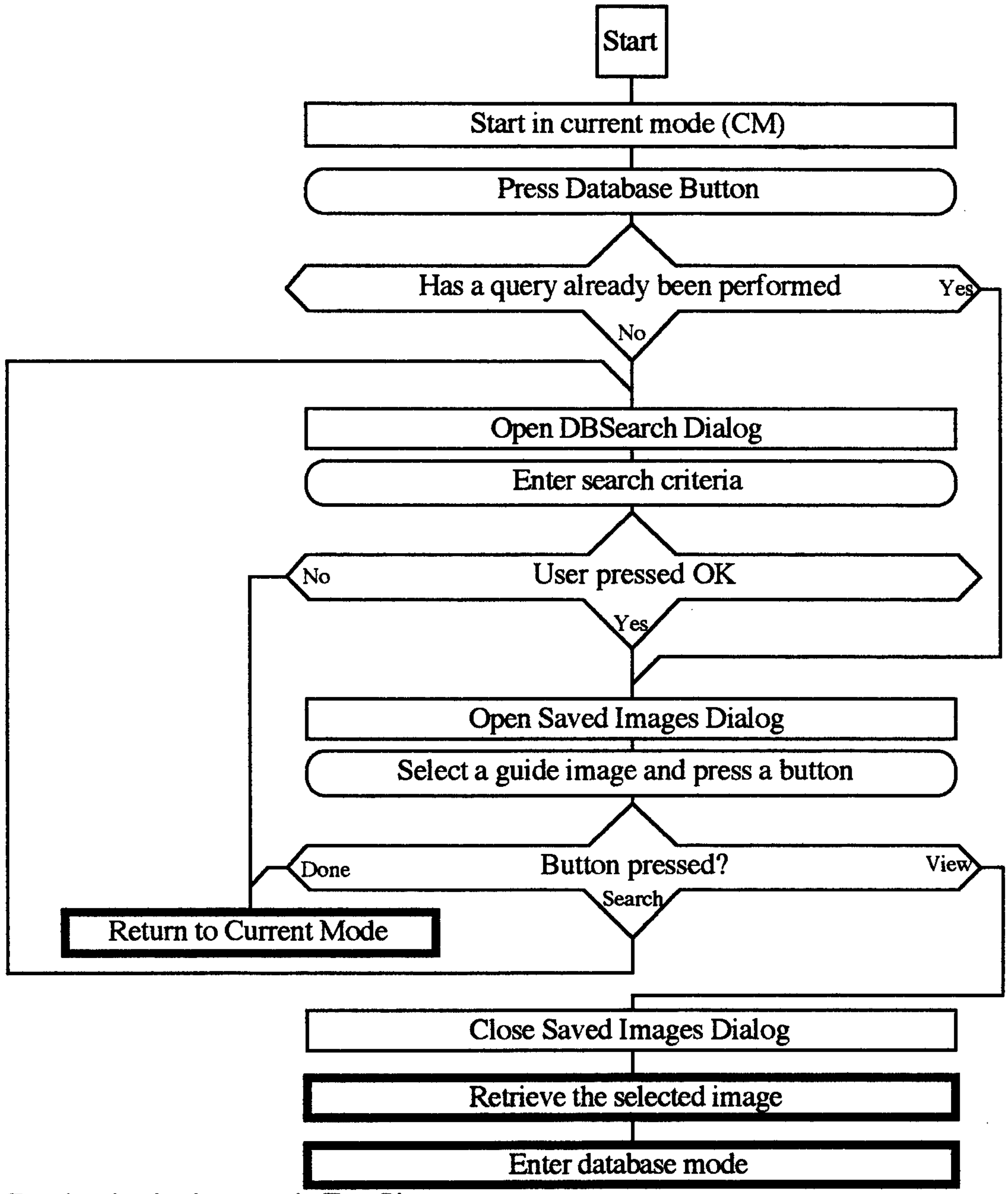
During any image review session (either guide or high magnification) the user can request a chat with a remote user. This is done by pressing the chat button. During this chat session the users will share the mouse and will be able to converse over the standard Macintosh sound channels. At the time of this writing the intended vehicle for implementing the chat is Open Transport (See section 2.7 for a description of Open Transport).



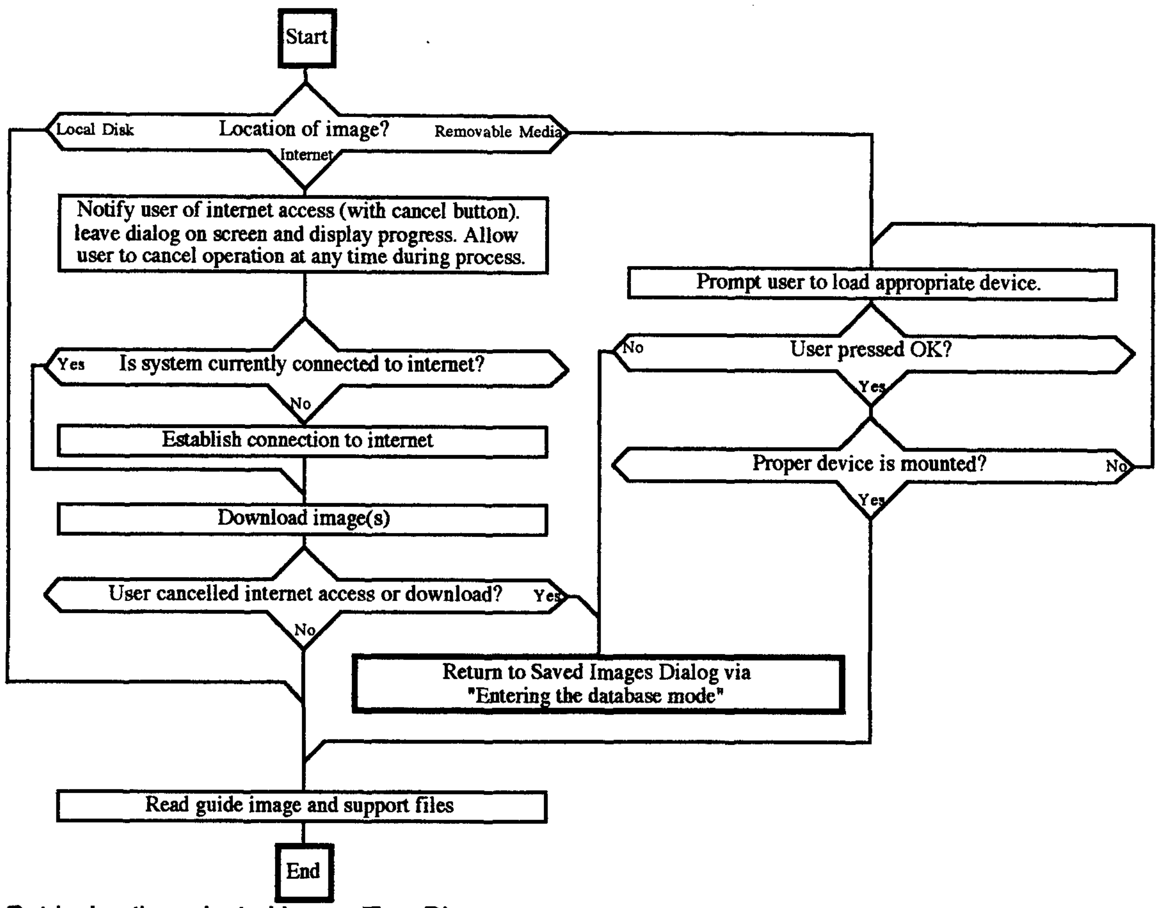
Chat Flow Diagram

3.9 Single Station - Database Mode

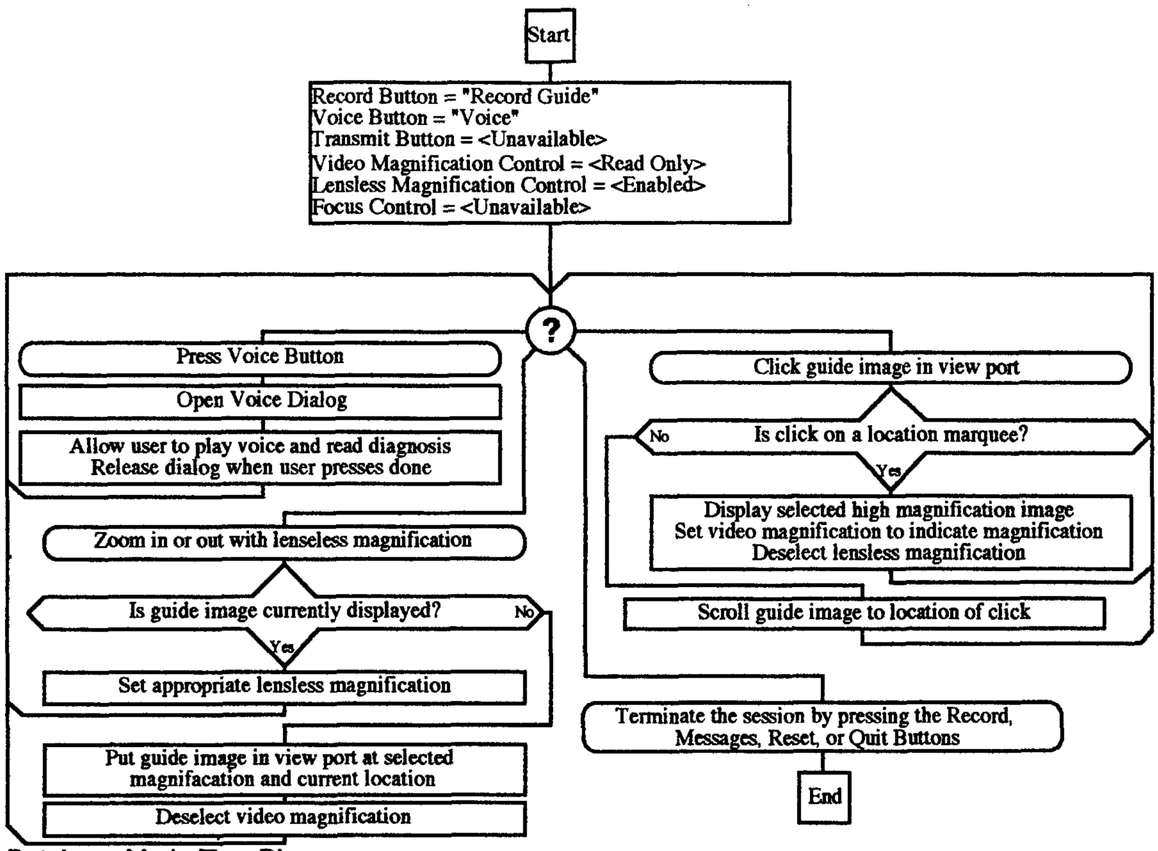
By pressing the database button the user will be given a screen in which to enter search criteria. If no criteria is entered then all available images will be retrieved. The user can enter any of the standard SQL (Structured Query Language) wildcards for search criteria. After entering the criteria and initiating the search the user will be taken to the saved images dialog and allowed to select from the guide images found for that query. Choosing a guide image will return the user to the main window in database mode.



Entering the database mode Flow Diagram

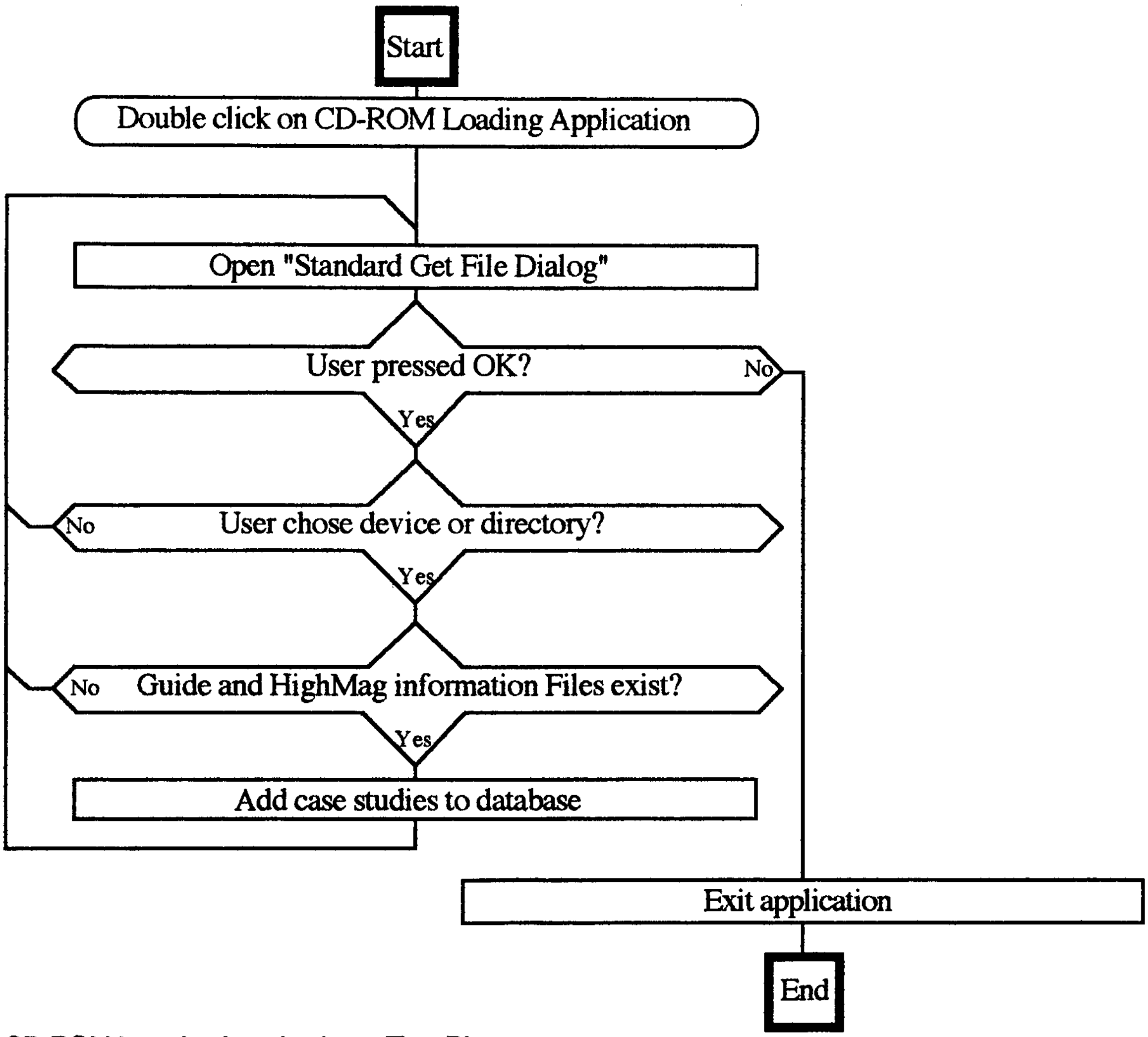


Retrieving the selected image Flow Diagram



Database Mode Flow Diagram

The Tele-Pathology Workstation will ship with a set of CD-ROM's containing public domain case studies which have been already incorporated into the database. Occasionally the user may be in a position to incorporate new CD-ROM's. In order to achieve this task the user will launch a utility application which facilitate the addition of new case studies into the database. The following flow diagram illustrates this procedure.



CD-ROM insertion into database Flow Diagram

In addition to inserting new removable media into the database the user will also need to move completed cases to removable media as the local hard disk fills up. The following flow diagram illustrates this task.

Moving the completed cases to removable media Flow Diagram

4 OBJECT IDENTIFICATION

The following C++ class objects have been identified as needed objects for the execution of the Tele-Pathology Workstation system. Each of the objects is defined in as much detail as is possible at the current time.

4.1 Naming Conventions

The following naming conventions apply to all members and methods in the Tele-Pathology Workstation project. All rules established in the Kensal C++ Coding Standards will apply where not specified.

Member Prefixes:

The prefix will denote the storage method of the member.

Type	Prefix
Handles	hnd
Pointers	ptr
Structures	its

Member Suffixes:

The suffix will denote the usage type of the member.

Type	Suffix
Buttons	Btn
Popup Menus	PU
Static Text	TX
Edit Text	TE
Field Text	TF
Tables/Lists	Tbl
Radio Buttons	Rad
Check Boxes	CBx

For a complete listing of all classes and their definitions see the TPW project code header files.

4.2 Object Summary

Object	Parent	Description
Kensal Foundation Class:		
KFCApp	CApp	A generic application object.
KFCBlockSmooth	NA	An object for performing JPEG block smoothing
KFCButton	CButton	A multi-purpose button which allows different special effects.
KFCColor	KFCServices	An abstract color services classes which is functionally derived by several service objects.
KFCColorGrayRGBtoYCC	KFCColorRGBtoYCC	A service object for Grayscale RGB to YCbCr colorspace conversion.
KFCColorGStoYCC	KFCColorYCCtoYCC	A service object for Grayscale to YCbCr colorspace conversion.
KFCColorRGBtoYCC	KFCColor	A service object for RGB to YCbCr colorspace conversion.
KFCColorYCCtoGS	KFCColorYCCtoYCC	A service object for YCbCr to Grayscale colorspace conversion.
KFCColorYCCtoRGB	KFCColor	A service object for YCbCr to RGB colorspace conversion.
KFCColorYCCtoYCC	KFCColor	A service object for No colorspace conversion.
KFCCommFTP	KFCCommInternet	A communication object for accessing the internet via FTP.
KFCCommHTTP	KFCCommInternet	A communication object for accessing the internet via HTTP.
KFCCommInternet	KFCCommunication	A communication object for accessing the internet.
KFCCommunication	NA	The virtual communication class.
KFCConfirm3StateDlg	CDialogDirector	A dialog box which will return one of three possible results (-1, 0, 1).
KFCConfirmDlg	CDialogDirector	A dialog box which will return one of two possible results (0, 1).
KFCConfirmListDlg	CDialogDirector	A dialog box which takes a pointer to a KFCStringList object and allows the user to select an item. The dialog returns the item number selected or 0 if cancelled.
KFCDebug	NA	A file containing helpful debugging routines (not an object).
KFCDesktop	CDesktop	A desktop object for floating window click sensing and menu bar hiding.
KFCErrorDlg	CDialogDirector	A dialog box for alerting the user of error conditions.
KFCExpanderBtn	CIconButton	An icon button which remembers the expansion value.
KFCFieldText	CDialogText	A text object for displaying static text in a field (box).
KFCFile	CDataFile	A generic file handler.
KFCFileDicom	KFCFileImage	The entry point for DICOM file management

KFCFileDicomDir	KFCFileDicomObject	An object for supporting the DICOMDIR file.
KFCFileDicomImage	KFCFileDicomObject	A DICOM image object
KFCFileDicomMetadata	KFCFileDicomTagList	A DICOM metadata object in a DICOM file
KFCFileDicomObject	NA	A generic DICOM object (List of Tags)
KFCFileDicomObjectList	CVoidPtrArray	A list of DICOM objects
KFCFileDicomTag	NA	A generic DICOM tag
KFCFileDicomTagImage	KFCFileDicomTag	A DICOM image tag
KFCFileDicomTagImplicit	KFCFileDicomTag	An implicit DICOM tag
KFCFileDicomTagList	CVoidPtrArray	A list of DICOM tags
KFCFileDicomTagSequence	KFCFileDicomTag	A DICOM tag sequence
KFCFileDicomTagSequenceItem	KFCFileDicomTagSequence	A DICOM tag sequence item
KFCFileGIF	KFCFileImage	A robust GIF file manager.
KFCFileGIFImage	N/A	A GIF Image in a GIF file.
KFCFileGIFImageList	CVoidPtrArray	A list of GIF Images derived as CPtrArray<KFCFileGIFImage>.
KFCFileImage	KFCFile	A generic image file handler.
KFCFileJPEG	KFCFileImage	A robust JPEG file manager.
KFCFileJPEGImage	N/A	A JPEG Image in a JPEG file.
KFCFileJPEGImageList	CVoidPtrArray	A list of JPEG Images derived as CPtrArray<KFCFileJPEGImage>.
KFCFileQuickTime	KFCFileSound	A sound file class for working with quicktime files.
KFCFileSound	KFCFile	A format free sound file class.
KFCFileTIFF	KFCFileImage	A robust TIFF file manager.
KFCFileTIFFImage	N/A	A TIFF Image in a TIFF file.
KFCFileTIFFImageList	CVoidPtrArray	A list of TIFF images within one file derived as CPtrArray<KFCFileTIFFTagList >.
KFCFileTIFFTag	N/A	A TIFF Tag member of the TIFF file.
KFCFileWave	KFCFileSound	A wave file management object
KFCFlexiblePICTGrid	CPICTGrid	A PICT grid with customizable selection mechanisms.
KFCGlobs	N/A	An object for providing some static functions and global variable initialization.
KFCGridScroll	CScrollPane	A scroll bar for scrolling according to a pre-defined grid dimension.
KFCHuffman	KFCServices	An abstract huffman object
KFCHuffmanDecode	KFCHuffman	An object for decoding huffman data
KFCHuffmanEncode	KFCHuffman	An object for encoding huffman data
KFCIdleChores	CChore	A class for portioning out time to tasks at idle time.
KFCImage	N/A	A virtual class for image management.
KFCImageDims	N/A	An object for storiing a standardized image description.

KFCImageGWorld	KFCImageHead	A GWorld object derived as KFCImageHead<GWorldPtr>.
KFCImageList	CVoidPtrArray	A list of images used by the KFCFlexiblePICTGrid.
KFCImagePane	CPanorama	An object which allows any kind of an image to be displayed and sense clicks and scroll.
KFCImagePICT	KFCImageHead	A PICT object derived as KFCImageHead<PicHandle>.
KFCImagePixMap	KFCImageHead	A PixMap object derived as KFCImageHead<PixMapPtr>.
KFCJPEGPipe	NA	A generic JPEG controller
KFCJPEGPipeComplex	KFCJPEGPipe	A JPEG controller for handling multiple pass coding
KFCJPEGPipeComplexEntropy	KFCJPEGPipeComplex	A JPEG controller for entropy multiple pass coding
KFCJPEGPipeEntropy	KFCJPEGPipe	A JPEG controller of entropy single pass coding
KFCMCU	KFCServices	An abstract object for performing Discrete Cosine Transforms.
KFCMCUExtract	KFCMCU	A DCT Extraction and quantization object.
KFCMCUInsert	KFCMCU	A DCT disassembler object.
KFCMCUInsertInterleaved	KFCMCUInsert	An interleaved DCT disassembler.
KFCNetwork	N/A	An object which connections to a network.
KFCPasswordText	CDialogText	A text object for entering passwords (the input characters are masked).
KFCProgressBar	CRectOvalButton	A progress bar object.
KFCPtrArray	CPtrArray	An array template to provide a few more features than CPtrArray.
KFCQuantizer	KFCServices	An abstract object for performing color quantization services.
KFCQuantizer1Pass	KFCQuantizer	A single pass quantizer.
KFCQuantizer1Pass3Color	KFCQuantizer1Pass	A single pass quantizer for RGB images
KFCQuantizer1PassDither	KFCQuantizer1Pass	A single pass quantizer for dithered images
KFCQuantizer2Pass	KFCQuantizer	A double pass quantizer.
KFCSample	KFCServices	An abstract color sampling service object.
KFCSampleDn	KFCSample	A down sampling object
KFCSampleDnFull	KFCSampleDn	A full down sampling object
KFCSampleDnFullSmooth	KFCSampleDnFull	A full down sampling object that performs smoothing.
KFCSampleDnH2V1	KFCSampleDn	A 2:1 horizontal and 1:1 vertical down sampling object.
KFCSampleDnH2V2	KFCSampleDn	A 2:1 horizontal and 2:1 vertical down sampling object.

KFCSampleDnH2V2Smooth	KFCSampleDnH2V2	A 2:1 horizontal and 2:1 vertical down sampling object with smoothing.
KFCSampleDnInt	KFCSampleDn	An arbitrary integral down sampling object.
KFCSampleUp	KFCSample	An up sampling object
KFCSampleUpFull	KFCSampleUp	A full up sampling object
KFCSampleUpH2V1	KFCSampleUp	A 2:1 horizontal and 1:1 vertical up sampling object.
KFCSampleUpH2V2	KFCSampleUp	A 2:1 horizontal and 2:1 vertical up sampling object.
KFCSampleUpInt	KFCSampleUp	An arbitrary integral up sampling object.
KFCServices	N/A	The base class of the service objects.
KFCSlider	CSubViewDisplayer	A generic slider control.
KFCSliderBar	CPictureButton	The buttons in the slider.
KFCSliderBtn	CIconButton	The bar in the slider.
KFCSliderTE	CDialogText	The active text box associated with a slider bar.
KFCSliderTX	CStaticText	The inactive text box associated with a slider bar.
KFCSliderThumb	CIconButton	The thumb in the slider.
KFCTable	CArrayPane	A table which allows command/item associations and colored items.
KFCTask	N/A	A virtual task class.
KFCTaskList	CVoidPtrArray	A list of tasks.
KFCTaskProg	KFCTask	A generic progress task. This class does not do any drawing.
KFCTaskRdComm	KFCTask	A communication task class for reading from a remote location into a local file.
KFCTaskSound	KFCTask	A task object for playing sound objects.
KFCGenericString	N/A	An object for providing some static string functions.
KFCUtitlities	N/A	Some generic and useful static functions.

Tele-Pathology Workstation Classes:

TPWAddressArrayPane	CArrayPane	An object for displaying the addresses list.
TPWAddresses	CVoidPtrArray	An object for holding an array of address entries.
TPWAddressEntry	NA	An individual address.
TPWApp	CApplication	Handle all command parsing and switching. Perform global instance management.

TPWConnectionsDlg	CDialogDirector	Used to enter connections information for the valid addresses in the transmit dialog.
TPWDatabase	N/A	A Database interface object.
TPWDatabaseQuery	N/A	The object which remembers and manages the users queries on the database.
TPWDatabaseSchema	N/A	An object which is responsible for the creation of the entire database structure.
TPWDBEnterDlg	CDialogDirector	For gathering image data on diagnosis.
TPWDBImagesDlg	CDialogDirector	Handles all file retrieval for viewing of saved images. Is also responsible for displaying thumbnail images on screen.
TPWDBSearchDlg	CDialogDirector	For entering database search criteria.
TPWFocus	CDialogDirector	The focus control for the TPW camera.
TPWGuideFWind	CFloatDirector	Displays the current region of interest during high magnification image creation at 20x.
TPWHelpDlg	CDialogDirector	The help dialog for the TPW.
TPWIdleChore	CChore	A class for portioning out time to tasks at idle time.
TPWImageDataFWind	CFloatDirector	A display of the image data in the current record.
TPWImagePort	KFCImagePane	The view port will handle displaying of the images as well as sensing mouse clicks and performing pan and scroll with the slide table.
TPWLoginDlg	CDialogDirector	The main login dialog at startup in a multiple user environment.
TPWMagGuide	CSubViewDisplayer	Handles all software magnifications of the guide image.
TPWMagVideo	CSubViewDisplayer	Performs all hardware high magnification magnifications as well as indicating the current magnification factor. Also indicates and receives the users location magnification request.
TPWMainWind	CDocument	All functions are initiated and handled through the Main Window. This object is responsible for setting the screen objects and switching to the appropriate methods for actions chosen by the user.
TPWMessagesDlg	CDialogDirector	Remote Users Messages Dialog.
TPWOverlay	CSubViewDisplayer	To change the overlay display.
TPWOverlayData	N/A	An object used to describe an individual high magnification image region.

TPWOverlayList	CSubViewDisplayer	A List of TPWOverlayData Objects.
TPWPreferencesDlg	CDialogDirector	Preferences setting dialog.
TPWSplash	CDialogDirector	The Splash screen which is displayed on screen briefly during launch.
TPWTaskProg	KFCTaskProg	The progress bar drawing class.
TPWTaskRdImage	KFCTask	The image file reading class which draws to the image port.
TPWTaskVoiceDlg	KFCTask	An object for updating progress information and button status during voice play.
TPWThumbnail	KFCImageGWorld	The object responsible for managing a thumbnail in the guide mosaic.
TPWTransmitDlg	CDialogDirector	Gathers the recipients address and sends the request to TPWCommunication.
TPWVoiceDlg	CDialogDirector	Performs all sound manipulation tasks. Is also responsible for attaining diagnosis transcription.

RC/21 Objects:

BADObject	RCObject
BRANCH	N/A
BTree	N/A
BTREE_CURSOR	N/A
B_POSITION	N/A
Column	RCObject
Database	RCObject
DBVALUE	N/A
DCE	N/A
Dictionary	N/A
DTE	N/A
Fieldmap	N/A
FIELDPAGE	N/A
FILTER	N/A
FilterSearchObject	SearchObject
IMPT	RCObject
IMPTE	N/A
IosTie	N/A
IosUnitbuf	N/A
NAME_ENTRY	N/A
NECESSARY_RELATION	N/A
RCObject	N/A
SearchObject	RCObject
streamoff	N/A
streampos	N/A
Table	RCObject
TASK	N/A
TOKEN	N/A
VALDESC	N/A
VALUE	N/A

RC/21 Add-in Classes:

RCColumnBLOB	Column	An object for implementing the movement of PixMap data into and out of the BLOB column.
RCSerialDatabase	Database	For implementation of the serialized database.
RCSerialTable	Table	For implementation of the serialized table.

TPW Database Importer Classes:

TPDApp	CApplication	The importer application class.
TPDImportInternet	N/A	The object responsible for importing internet information.
TPDImportLocal	N/A	The object responsible for importing local information.
TPDiNetPrefs	CDialogDirector	The Internet importing preferences dialog.
TPDMain	CSaver	An object to provide a main interface derived from CSaver<CCollaborator>.

5 FILE FORMATS

The following outlines the basic usage of each file type and generally outlines the proposed file format.

5.1 Session Specific Files

The following outlines all of the possible file formats during a slide analysis session. A session is distinguishable as a slide number. Since each of the sessions have separate slide numbers and all session specific files for a given slide number are stored in a directory which has the same name as the slide number, it is not necessary to put the slide number in the file names.

The session analysis scenarios have been sub-divided into two major categories: DICOM and non-DICOM sessions. Although it is not foreseen at this time that the application will support non-DICOM analyses it is nevertheless mentioned due to it's previous design and current existence within the TPW application. Several of the TPW session specific files can be used in either scenario with equal functionality. These common files can be found at the end of this section. Each of the file names which utilizes a file name extension will implement the extension to indicate the files data type. In several cases the data type has many possibilities. Wherever the data type of the file is flexible the options will be noted. In the File Format sections the identification of a tab character is seen as a "<T>" and a return character as a "<R>."

5.1.1 DICOM analyses

The primary distinction in a DICOM analysis session is the presence of the DICOMDIR file. This file will be present at a root directory location and available for all DICOM analysis sessions. In a given session the image files will be contained in a single directory which has the same name as the slide number.

5.1.1.1 Guide Image

The guide image file stores the guide image only. This file will be typically stored as JPEG compressed data but may optionally be 24 or 15 bit RGB data. The guide image will contain all of the guide specific data needed to position the image within a slide representation and scale the image appropriately. See the Image Labs "Kensal DICOM File Specifications" for further discussion of the necessary tags and layout of this file. In order to accommodate a rapid full screen display of the image during the retrieval of the session it will also be necessary to store a 500x1000 thumbnail of the guide in the same directory. It is not necessary however to store the patient and excession information in this thumbnail as it will all be contained in the guide. The "DICOMDIR" file will contain all of the appropriate file names in the DICOM scenario.

<u>File Name:</u>

"00000000"

<u>File Type:</u>

Tag formatted DICOM.

5.1.1.2 High Magnification Image

The high magnification images will also contain the pertinent tags for there proper placement and scaling. The "DICOMDIR" file will contain all of the appropriate file names in the DICOM scenario.

<u>File Name:</u>

"00000001"

Examples:

"00000001" High magnification image at region 1

"00000002" High magnification image at region 2

File Type:

Each high magnification image will be stored by itself in a DICOM file. Their will be one file for each location where a high magnification picture was taken.

5.1.2 Non-DICOM analyses

5.1.2.1 Guide Image

The guide image file stores guide images, the thumbnail image and any color lookup tables which will be necessary.

File Name:

"GUIDE.TIF"

File Type:

The guide image file will contain both the scanned guide image at 10x and the thumbnail image reduced to 256x128. Each image will be saved in Tiff Compressed format. The file header will contain adequate information for their retrieval.

5.1.2.2 High Magnification Image

File Name:

"HM" + *location num.* + ".TIF"

Examples:

HM01.TIF High magnification image at region 1

HM02.TIF High magnification image at region 2

File Type:

Each high magnification image will be stored by itself in a Tiff Compressed file. Their will be one file for each location where a high magnification picture was taken. The location and magnification of the image can be reconciled through the regions file.

5.1.2.3 Image Data

Each image will have an associated line in the data file. The Slide number, Start Date, and Finish Date are stored automatically. The purpose of the data files is to perform similar services as those of the DICOMDIR file.

File Name:

"GD.TXT" / "HM.TXT"

File Type:

Image data is stored in a text file in tab delimited format.

File Format:

"GD.TXT"	"HM.TXT"
Folder<T>	Slide_No<T>
Slide_No<T>	Prefix<T>
Prefix<T>	Suffix<T>
Suffix<T>	Stain<T>
Stain<T>	Record_Dt<T>
Start_Dt<T>	Record_Tm<T>
Finish_Dt<T>	Sequence<T>
MD_LName<T>	Magnification<T>

MD_FName<T>
MD_MName<T>
Institution_Name<T>
Institution_City<T>
Institution_State<T>
Topography<T>
Morphology<T>
Etiology<T>
Function<T>
Disease<T>
Procedure<T>
Occupation<T>
Snomed<T>
Scale<T>
XOffset<T>
YOffset<T>
Technician_LName<T>
Technician_FName<T>
Technician_MName<T>
Media<R>

Scale<T>
XPosition<T>
XPosition<T>
Technician_LName<T>
Technician_FName<T>
Technician_MName

5.1.3 Regions

<u>File Name:</u>

"REGIONS.TXT"

<u>File Type:</u>

The regions file contains a comprehensive list of all location/magnifications visited, requested and saved during the entire session. The file is a simple text file in tab delimited format. The format is given below. The identifier column identifies the action (V = Visit, R = Request, S = Saved). The X and Y locations are in micro meters. The location number indicates the sequence in which the events occur. Notice that visits get decremented while requests and saves get incremented.

<u>File Format:</u>

<u>Header:</u>

TPW version<T>Slide No.<T> Guide Offset (Microns) Hor.<T> Vert.<R>

<u>Detail:</u>

Identifier<T>
Location Number<T>
Magnification<T>
X Location<T>
Y Location<T>
Date and Time<T>
User<R>

<u>File Example:</u>

Note - This is not a real life example.

```
1.10        12345  0      0
V   -1      200    34288  20898  95.09.13-13:43  Smith@Mayo
R    1      400    5543   5432   95.11.13-15:43  Smith@Mayo
S    1      100    876    9965   95.11.14-09:43  Smith@Mayo
```

5.1.4 Voice

The voice file stores a single digitized voice. Since the same location can record voice from both the local and remote user and the slide examination session can span multiple iterations it becomes necessary to be able to delineate the files. The voice file can be saved as either a Quicktime file (.MOV) or a Wave file (.WAV)

File Name:

["GD_" | "HM" + *location* + "_"] + [*user* ("L" | "R") + iteration | "DG"] + ".XXX"

Examples:

GD_R01.WAV	The remote users 1st iteration for the guide image
HM08_L02.MOV	The local users 2nd iteration for the high magnification image at region 8
GD_DG.WAV	The pathologists diagnosis for the guide image
HM02_DG.MOV	The pathologists diagnosis for the high magnification at region 2

File Type:

This is a standard QuickTime™ of Microsoft Wave file storing simply the digitized voice.

5.1.6 Diagnosis

File Name:

"DIAGNOS.TXT"

File Type:

The diagnosis file will contain a transcribed version of each voice file given during the transcription mode.

File Format:

Diagnosis Text (with optional embedded carriage returns) <R>
["Region" + space + region # + ":" < Return >
Region Specific Diagnosis Text (with optional embedded carriage returns) <R>]
[more locations...]

File Example:

```
Some text which is associated with the entire slide analysis session.

Possibly with some embedded return characters.

Region 7:
The above string will flag the TPW that this text is associated with a specific location, in this case
location number 7.

Region 12:
The file can contain as many locations as necessary but should only include each location once.
```

5.2 Global Files

5.2.1 Messages

The information in the messages file will be tab/return delimited. Messages will be indicated by a code: A = Analysis Requested, L = Image Locations Requested, I = Images Received, D = Diagnosis Received. An asterisk will indicate that the message has been paused. An entry for the resolution will indicate that the message has been addressed.

<u>File Name:</u>

TPW_Messages

<u>File Type:</u>

A standard text file will hold all of the messages for a predetermined period of time. An archive of messages files will be kept indefinitely.

<u>File Format:</u>

Message type & pause indicator<T>
Date and Time Received<T>
Sender<T>
Receiver<T>
Resolution Date and Time<R>

<u>File Example:</u>

A	95.09.23-12:14	Jones@Mayo	Warner@J.Hopkins	95.09.23-14:11
L*	95.09.23-13:33	Smith@PCH	Hanks@J.Hopkins	
I	95.09.23-13:33	Jones@Mayo	Warner@J.Hopkins	

5.2.2 Addresses

The addresses file contains all of the saved addresses for the system. Each entry will contain an identifier as to its base connection type (either D or F for Direct or FTP respectively) and will be followed by the appropriate information according to the connection profile. At present the only connection profile identified is Direct over ISDN. As more connection profiles are incorporated a standard format for them will be adopted. If the profile requires a password the password will be encrypted.

<u>File Name:</u>

TPW_Addresses

<u>File Type:</u>

The file is standard text in tab/return delimited format.

<u>File Format:</u>

ISDN:
Connection Name<T>
Connection Type<T>
Hardware<T>
Phone Number<T>
Login<T>
Password<T>
Owner<R>

<u>File Example:</u>

Jones@Mayo	D	ISDN	1-602/555-8877	WarnerJH	^%Ì™	Warner
Smith@PCH	D	ISDN	1-602/555-7576	Warner	^ËÁÏ	Warner
Erics@Mayo	D	ISDN	1-602/555-8877	HanksJH	ÌÏ‰	Hanks
Timms@PCH	D	ISDN	1-602/555-7576	Hanks	·°‡·	Hanks

5.2.3 Preferences

The preferences file will contain the preference settings as well as the pertinent information to implement the settings. Since the system will be used by multiple users who will log in with a user name the preferences will be user specific if the preferences file indicates multi user (i.e. no login/password stored). The brackets ('[' and ']') indicate the optional syntax if multi user preferences is in use.

<u>File Name:</u>

TPW_Preferences

<u>File Type:</u>

A standard text file.

<u>File Format:</u>

"Mode:" <T>Single|Multi<R>

["Login:" <T>Login Name<R>

"Password :" <T>Password<R>]

"Messages:"<T>Messages Boolean[<Comma>User 1<T>
Messages Boolean<Comma>User 2<T>
... Messages Boolean<Comma>User n]<R>

"Locations:"<T>Number of Windows<T>Locations Boolean[<Comma>User 1<T>
Locations Boolean<Comma>User 2<T>
... Locations Boolean<Comma>User n]<R>

Window 1 Name<T>
Left<Comma>Top<Comma>Right<Comma>Bottom[<Comma>User 1<T>
Left<Comma>Top<Comma>Right<Comma>Bottom<Comma>User 1<T>
... Left<Comma>Top<Comma>Right<Comma>Bottom<Comma>User n]<R>

... Window n Name<T>
Left<Comma>Top<Comma>Right<Comma>Bottom[<Comma>User 1<T>
Left<Comma>Top<Comma>Right<Comma>Bottom<Comma>User 1<T>
... Left<Comma>Top<Comma>Right<Comma>Bottom<Comma>User n]<R>

"Transcriptionist:"<T>Transcriptionist Name[<Comma>User 1<T>
Transcriptionist Name <Comma>User 2<T>
... Transcriptionist Name <Comma>User n]<R>
"LUTs:"
"TC217_1"<T>Channel<T>Threshold<T>Offset<T>Gain<T>Gamma <R>
"TC217_2"<T>Channel<T>Threshold<T>Offset<T>Gain<T>Gamma <R>
"TC217_3"<T>Channel<T>Threshold<T>Offset<T>Gain<T>Gamma <R>
"RL4000P_1"<T>Channel<T>Threshold<T>Offset<T>Gain<T>Gamma <R>
"RL4000P_2"<T>Channel<T>Threshold<T>Offset<T>Gain<T>Gamma <R>

<u>File Example:</u>

Single User:

Mode:	Single	
Login:	Jones	
Password:	ÏÁÊÂÁÏ	
Messages:	TRUE	
Locations:	2	TRUE
Guide	35,67,291,323	
Data	950,876,1280,1024	
Transcriptionist:	Chenry	
LUTs:		

TC217_1	1	511	127	14.0	0
TC217_2	1	511	127	14.0	0
TC217_3	1	511	127	14.0	0
RL4000P_1	1	511	127	14.0	0
RL4000P_2	1	511	127	14.0	0

Multi User

Mode:	Multi				
Messages:	TRUE,Warner				FALSE,Hanks
Locations:	2	TRUE,Warner			FALSE,Hanks
Guide	35,67,291,323,Warner				0,0,0,0,Hanks
Data	950,876,1280,1024,Warner				950,5,1280,153,Hanks
Transcriptionist:	CHenry,Warner				PDempsy,Hanks
LUTs:					
TC217_1	1	511	127	14.0	0
TC217_2	1	511	127	14.0	0
TC217_3	1	511	127	14.0	0
RL4000P_1	1	511	127	14.0	0
RL4000P_2	1	511	127	14.0	0

6 Errors, Warnings, and Messages

The following section outlines the primary user messages which will occur throughout the use of the system. This is not presented as a comprehensive list of all possible messages as more will undoubtedly be uncovered during the implementation phase. Rather this provides a broad outline of the kinds of anomalous situations in which the user may find themselves. Variables which will be substituted at time of use are shown in braces.

6.1 User Errors

Message	Reference Section
Slide number {slide no} already exists. Please try again.	3.1
Slide number {slide no} is too [short l long]. Please try again.	3.1
Sorry, There is no slide loaded in the microscope.	3.1, 3.3

6.2 System Warnings

Message	Reference Section
The {mode} is incomplete and will be saved as a paused message.	N/A

6.3 General Messages

Message	Reference Section
Before continuing the chat some images need to be transferred. You will be notified once transfer is complete.	3.8

6.4 User Requests

Each request string is given below with the button titles in parentheses. The default button is underlined.

Message	Reference Section
Messages are pending do you want to respond? (<u>Yes</u>, No)	3
Are you sure you want to pause the {mode} from {user}? (<u>Yes</u>, No)	3
Please put slide number {S#} in the microscope and press OK. (Cancel, <u>OK</u>)	3.3
Are you sure you want to terminate the chat with {user}? To pause the chat press the pause button. (Pause, Cancel, <u>OK</u>)	3.8
User {user} would like to chat. Clicking OK will pause your current {mode}. (Cancel, <u>OK</u>)	3.8

7 DATABASE

This section describes the structure and intended function of the Tele-Pathology Workstation database.

7.1 Description

The TPW database has been developed over a core of objects developed by the Vermont Database Corporation. This database core is known as the RC/21 database library. The RC/21 is a full featured relational database with BLOB (Binary Large Object), referential integrity, transaction processing, and other necessary TPW features. In order to customize the RC/21 to suite the TPW several derived object have been developed as can be seen above. In addition, a table is created with each RCSerialDatabase whose purpose is to implement the serial data types.

7.2 Schema

The following section describes the structure of the database as it shall be implemented for the TPW. Additional tables and/or columns may be necessary in the future as need dictates during actual usage.

Data Types:

Serial	An indexed integer column which is set up to automatically increment in value.
Integer	4 byte whole numbers.
String	Alpha numeric information of any length.
Real	8 byte floating point numbers.
Blob	ASCII character data of any size.
Date	Internal integer values stored as the number of days since 1 January 1904.
Time	The time of the day stored in seconds from midnight. A Real value.

Table/Column	Data Type	Description
Images Table:		The images table is designed to hold all guide image specific information. The images table can be seen as the primary table in the database.
image_id	Serial	images table primary key.
device_id	Integer	foreign key to the devices table.
institute_id	Integer	foreign key to the institutes table.
md_id	Integer	foreign key to the MD table.
operator_id	Integer	foreign key to the operators table.
folder	String	folder name of the image and it's associated files (8 characters or less ISO9660).
slide_no	String	7 digit medical slide number.
file_name	String	file name of the image file.
prefix	String	a slide number prefix.
suffix	String	a slide number suffix.
stain	String	a code indicating the stain used on the slide.
start_dt	Date	the date which the guide image was recorded or the case was started.
finish_dt	Date	the date which the case was finished.
topography	String	SNOMED topography code.
morphology	String	SNOMED morphology code.
function	String	SNOMED function code.
etiology	String	SNOMED etiology code.
living_org	String	SNOMED etiology: living organism code.
chem_etc	String	SNOMED etiology: chemicals, etc. code.
phys_agents	String	SNOMED etiology: physical agents code.
occupation	String	SNOMED occupation code.
social	String	SNOMED social context code.
disease	String	SNOMED disease code.
procedure	String	SNOMED procedure code.

general	String	SNOMED general code.
description	String	full text description of the image.
scale	Real	the scale of the image in microns per pixel.
xoffset	Integer	the offset from the label of the slide to the center of the image in microns (slide label held in right hand).
yoffset	Integer	the offset from the top of the slide to the center of the image in microns (slide label held in right hand).
thumbnail	Blob	thumbnail stored as a flattened PixMap (256x128).
results_name	String	file name of the results file.
Devices Table:		All of the individual devices which are recognized by the system will have an entry in the devices table. This makes it quite simple in the future to implement such features and CD-ROM burning utilities.
device_id	Serial	devices table primary key.
media_type	Integer	a 4 character mnemonic indicating the type of the media ('cdrm', 'locl', 'inet', ...).
name	String	the short name of the device.
path	String	the full path of the device.
create_dt	Integer	the date of device as applicable.
icon	Blob	an icon representing the device. Stored as a CIcon.
Institutes Table:		General information about the medical institution which originates the slides. This is not meant to be a comprehensive detailing of the institute.
institute_id	Serial	institutes table primary key.
name	String	institutes full name.
address	String	1st address line.
address_opt	String	2nd address line (suite no., bldg. no., etc.).
city	String	full city name.
state	String	2 character state abbreviation.
zip	String	postal code.
country	String	full country name (if blank USA assumed).
MD Table:		Information about the professionals who are primarily responsible for the diagnosis of the slides.
md_id	Serial	MD's table primary key.
fname	String	given name of MD
mname	String	initials of MD
lname	String	surname of MD
salutation	String	appropriate leading salutation (Mr., Mrs., Ms., etc.).
position	String	applicable position (Physician, Resident, etc.).
title	String	applicable title (President, Chief Resident, etc.).
credentials	String	credentials and entitlements (MD, D.D.E., etc.).
address	String	1st address line.

address_opt	String	2nd address line (suite no., bldg. no., etc.).
city	String	full city name.
state	String	2 character state abbreviation.
zip	String	postal code.
country	String	full country name (if blank USA assumed).

Operators Table: Information about the individuals who are responsible for image recording.

operator_id	Serial	operator tables primary key.
fname	String	given name of operator.
mname	String	initials of operator.
lname	String	surname of operator.

HighMags Table: A table for the storage of the high magnification image logistical data.

highmag_id	Serial	highmag tables primary key.
image_id (1)	Integer	foreign key to images table.
file_name	String	file name of the image file.
sequence (1)	Integer	highmag image sequence within the guide images set of highmags.
record_dt	Date	date of recording the image.
record_tm	Time	time of recording the image.
magnification	Integer	ocular magnification of recording.
scale	Real	the scale of the image in microns per pixel.
xposition	Integer	horizontal position of the center point of the image with respect to the guide image in microns.
yposition	Integer	vertical position of the center point of the image with respect to the guide image in microns.
width	Integer	height of the image in pixels.
height	Integer	width of the image in pixels.

Glossary of Terms

Enabled - A control or screen object which is capable of being manipulated by the user.

Disabled - A control or screen object which is incapable of being manipulated by the user but which is still visible. The visibility will undoubtedly be in some sense indicating that the object is inaccessible.

Unavailable - A control or screen object which is not visible to the user and is further not functional.

Mode - A description of the users operation capabilities according the work that the user is attempting to perform.

Scenario - A description of a typical task that the user might wish to perform. A Scenario will often times be comprised of many Modes.

Flow Diagram - A scenario in graphical terms.

Schema - A description of a databases structure at the table, column and relationship level

Table - A database entity of primary concern. A place to hold rows of information pertaining to the same subject.

Column - A single definable database table attribute.

Row - A database entity instantiation.

Relationship - A defined "joining" of database entities.

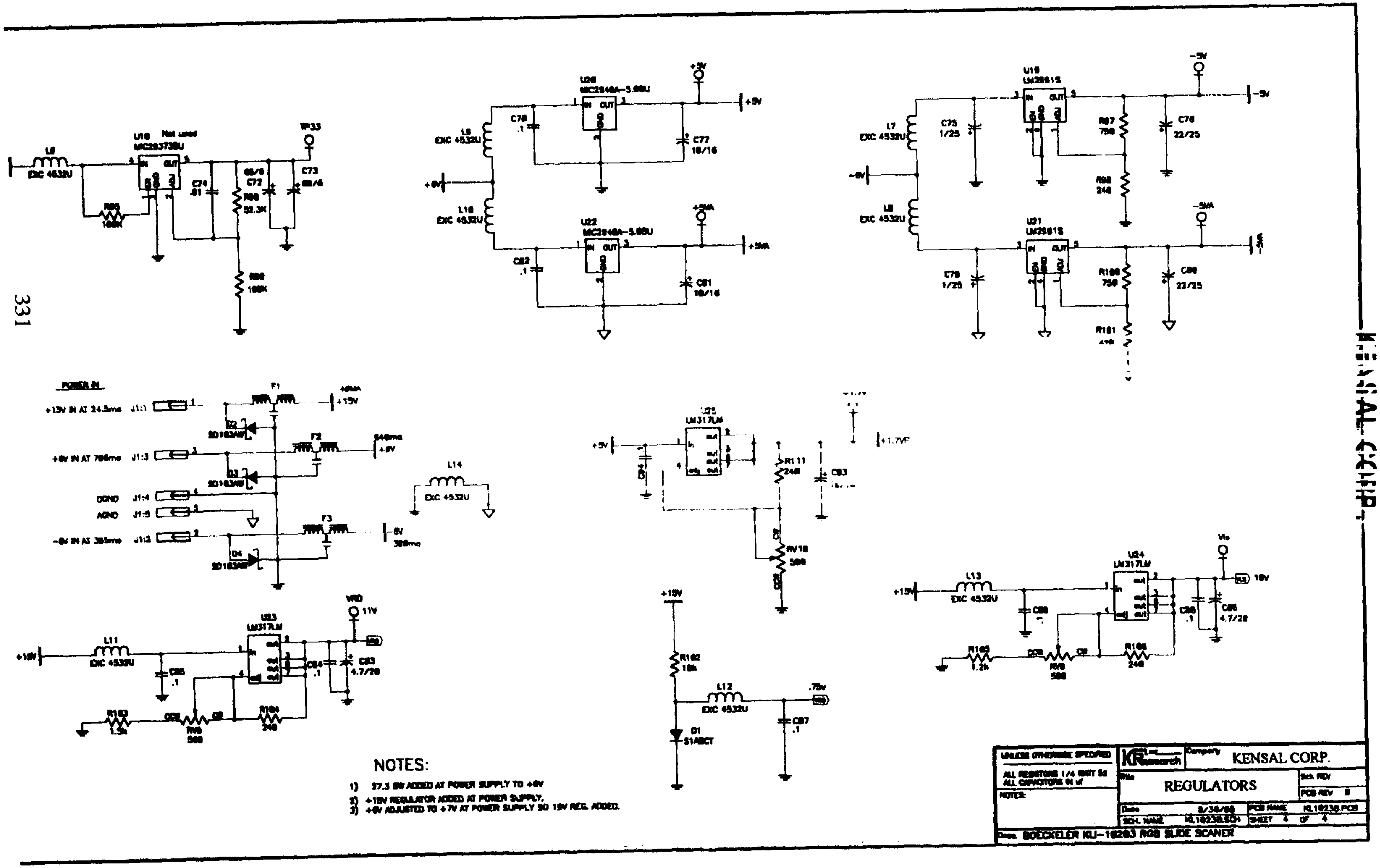
APPENDIX C

RAW DATA
FROM
1996 LUKE/MAYO TELEPATHOLOGY STUDY

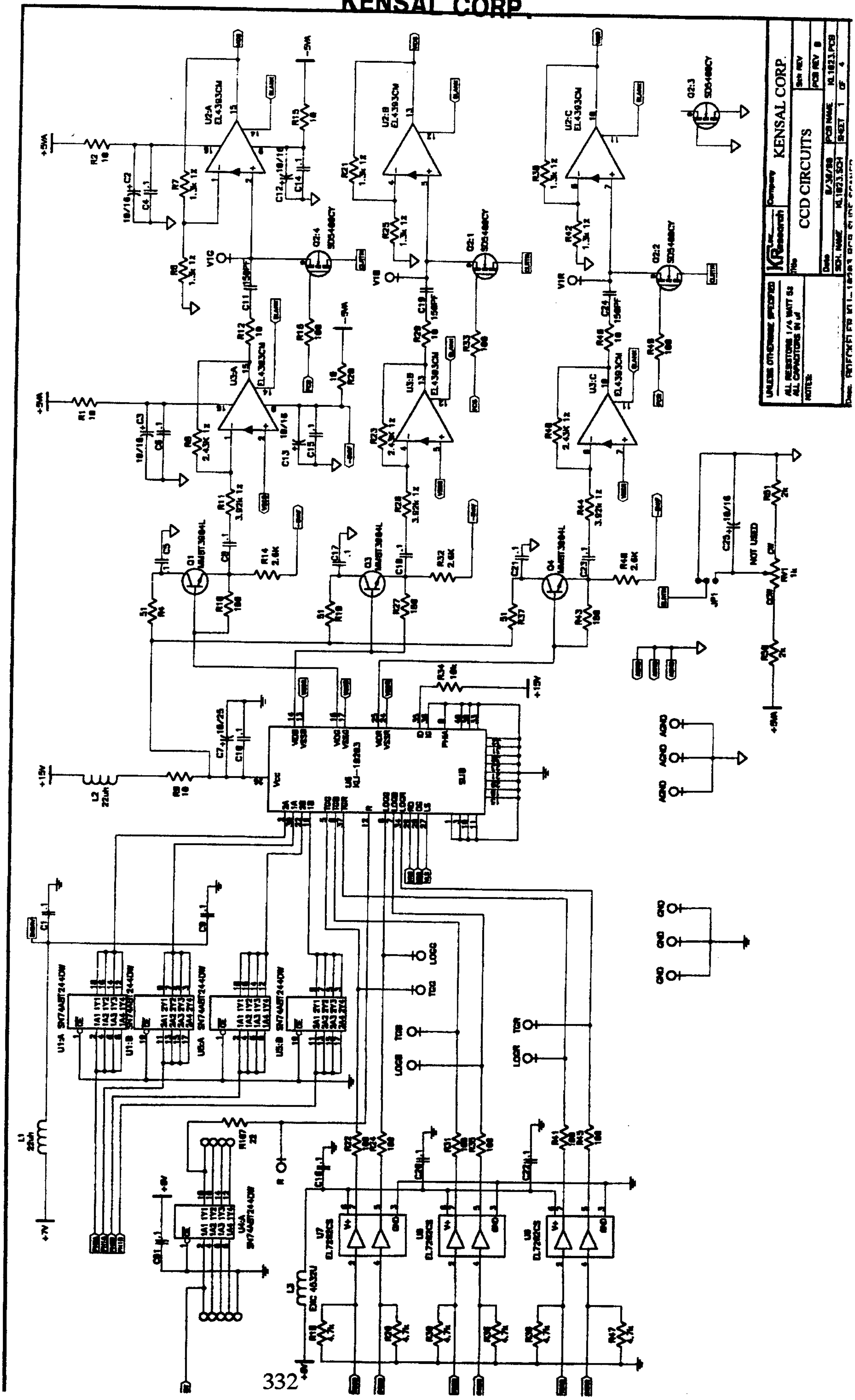
Slide	Source	Diagnosis from Workstation Images	Actual Diagnosis from Patient Files	Diagnosis from?	AOI's requested
501000	Mayo	High grade dysplasia, carcinoma insitu	Epidermoid carcinoma in situ (Bowen's Disease)	5th high mag (20x)	10
502000	Mayo	Immunoperoxidase stain needed for diagnosis;	Metastatic adenocarcinoma	N/A	27
		(guessing melanoma or undifferentiated carcinoma)			
503000	Mayo	No diagnosis given	Inverting epithelial papilloma (sino-nasal type)	N/A	8
503100	Mayo	Benign neoplasm, probably an inverted papilloma	Inverting epithelial papilloma	7th high mag (20x)	10
504000	Mayo	Malignant lymphoma	Malignant Lymphoma	9th high mag (40x)	13
504100	Mayo	Same as previous case	Same as previous case	No diagnosis given	8
505000	Mayo	Medication induced "black thyroid"	"Black thyroid"	7th location (40x)	7
505100	Mayo	Black thyroid due to treatment with monocycline	"Black thyroid"	13th location (40x)	15
506000	Mayo	Carcinoid tumor	Atypical bronchial carcinoid tumor	28th location (20x)	35
506100	Mayo	Carcinoid tumor	Atypical bronchial carcinoid tumor	4th location (4x)	21
506200	Mayo	Bronchial carcinoid tumor	Atypical bronchial carcinoid tumor	12th location (20x)	21
507000	Mayo	Intracystic (intraductal) adenocarcinoma, low grade	Intracystic papillary carcinoma	15th location (40x)	16
507100	Mayo	Neuroendocrine differentiation suggested, no diagnosis given	Intracystic papillary carcinoma	9th location (10x)	19
507200	Mayo	Intracystic (intraductal) well differentiated adenocarcinoma of ductal origin	Intracystic papillary carcinoma	16th location (40x)	16
508000	Mayo	Hepatatic capsular hamartoma (multiple)	Hamartomatous capsular scar	30th location (20x)	34
509000	Mayo	Lobular carcinoma insitu	Multifocal lobular carcinoma in situ	32nd location (10x)	33
510000	Mayo	(Only a Scout Image in this file; no voice files)	Alveolar Proteinosis	N/A	0
511000	Mayo	Primary biliary cirrhosis, Stage 3	Chronic Hepatitis (Syndrome of primary biliary cirrhosis, Stage 3)	29th location (20x)	29
512000	Mayo	Benign lyomyoma with nuclear atypia	Bizarre Atypical Leiomyoma	18th location (20x)	20
513000	Mayo	Lobular carcinoma insitu, superimposed on pre-existing lesion of fibroadenoma	Lobular carcinoma in situ arising in a fibroadenoma	19th location (20x)	25
514000	Mayo	Lymphangiomyomatosis	Lymphangiomyomatosis	11th location (20x)	12
515000	Mayo	Venus outflow obstruction and secondary massive zonal necrosis	Venus outflow obstruction with zonal necrosis	13th location (40x)	13
516000	Mayo	Granulosa cell tumor of the ovary	Granulosa Cell Tumor	16th location	16
517000	Mayo	Epithelioid endothelium	Epithelioid Hemangioendothelioma	20th location	20
518000	Mayo	Cirrhosis secondary to alpha-1 antitrypsin	Cirrhosis with hemosiderosis secondary to alpha-1 antitrypsin deficiency	16th location	16
519100	Mayo	Benign adenofibromena cytoplasia	Benign Fibroglandular Prostatic tissue	34th location	34
520000	Mayo	Subacute and chronic prostatitis	Chronic Prostatitis, nagative for malignancy	28th location	29
601000	Luke	Fibroadenoma	Fibroadenoma	40th location (40x)	40
602000	Luke	Benign lesion; fibrocystic change of the breast	Fibrocystic change	9th location (10x)	11
603000	Luke	Adenomatus nodule	Adenomatous nodule	11th location (20x)	11
609000	Luke	Seborrhoid keratosis, inflamed	Seborrheic keratosis	7th location (no mag)	7
610000	Luke	Dermatomycosis	Dermal fibrosis	5th location (40x)	5
611000	Luke	Hemorrhoids	Anorectal polyp	6th location (no mag)	6
612000	Luke	Ovarian stromal cell tumor fibroma	Ovarian tumor	8th location (10x)	8
613000	Luke	Follicular carcinoma	?	6th location (no mag)	6
614000	Luke	Malignant melanoma	Malignant melanoma	16th location (no mag)	16
615000	Luke	Adenoma	Nodular sclerosing adenosis	6th location (no mag)	6
616000	Luke	Reactive lymphoid hyperplasia	Lymphoid reactive folicular hyperplasia	7th location (no mag)	7
617000	Luke	Occifying Fibroma (Epulis)	Epulis hyperplasia (gingival mass in canine specimen)	14th location (40x)	14
618000	Luke	Well differentiated prostatic adenocarcinoma, grade 1	Prostatic adenocarcinoma	8th location	8
619000	Luke	Occifying fibroma, epulis	Epulis hyperplasia (canine specimen)	9th location	9
620000	Luke	Fat necrosis of a reaction to a foreign body (i.e. silicone)	?	6th loction	

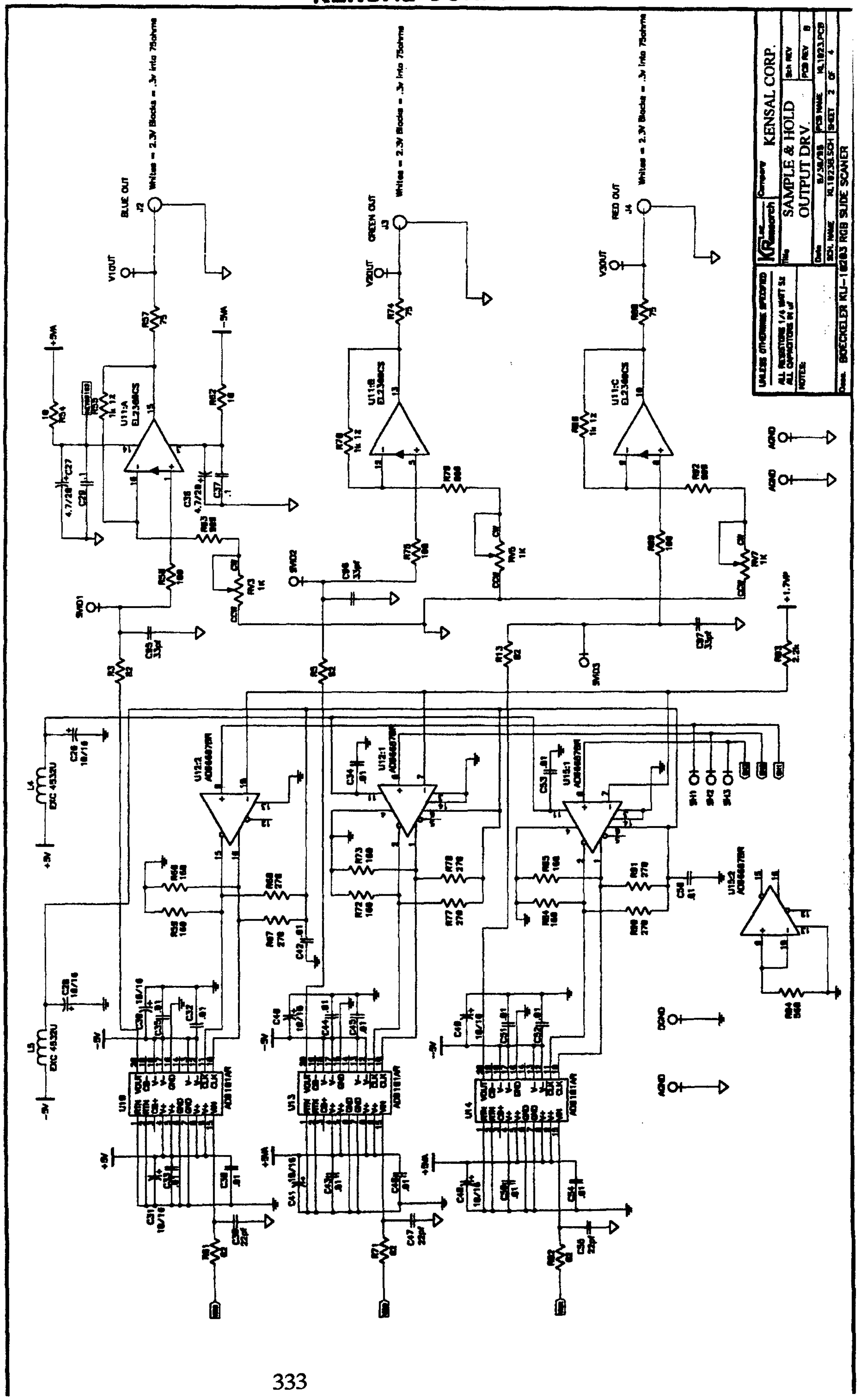
APPENDIX D

TSS SUPPORT DOCUMENTS

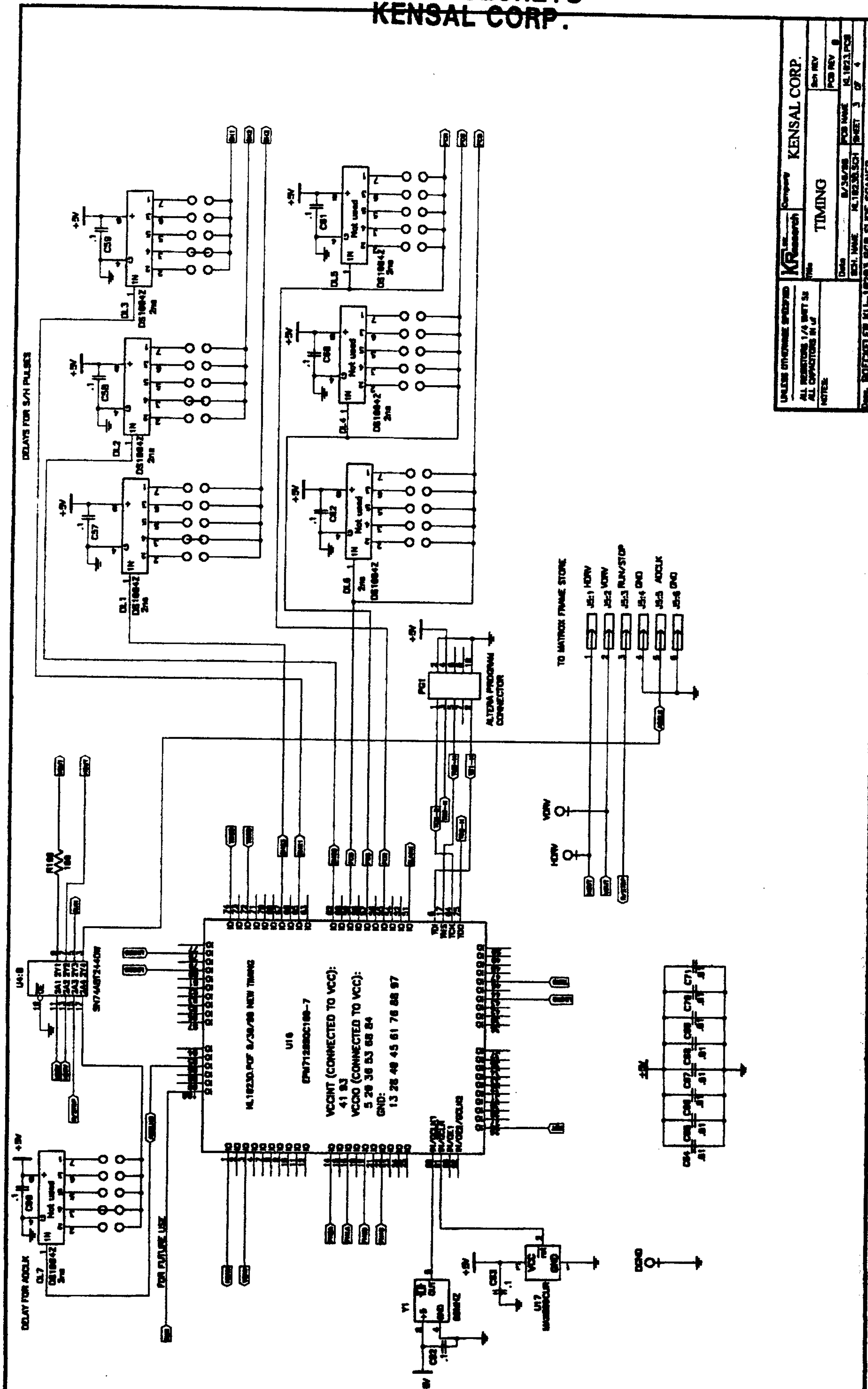

U18 Not used MC29373BU
L6 EXIC 4532U
R95 100K
C74 .01
R96 82.3K
C72 68/6
C73 68/6
TP33
R98 100K
U20 MIC2940A-3.8BU
U22 MIC2940A-3.8BU
L9 EXIC 4532U
L10 EXIC 4532U
C78 .1
C82 .1
C77 10/16
C81 10/16
+5V
+5VA
+6V
U19 LM2991S
U21 LM2991S
L7 EXIC 4532U
L8 EXIC 4532U
C75 1/25
C79 1/25
R97 750
R98 240
R100 750
R101 240
C76 22/25
C80 22/25
-5V
-5VA
-6V
POWER IN
+13V IN AT 24.5ma J1:1
+8V IN AT 786ma J1:3
DGND J1:4
AGND J1:5
-8V IN AT 385ma J1:2
F1
F2
F3
D2 SD103AW
D3 SD103AW
D4 SD103AW
+15V
+8V
-8V
L14 EXIC 4532U
U25 LM317LM
R111 240
C83
RV16 500
+1.7VF
U23 LM317LM
L11 EXIC 4532U
C85 .1
C84 .1
C83 4.7/20
VRD 11V
R103 1.3k
RV8 500
R104 240
+19V
R102 10k
L12 EXIC 4532U
D1 S1ABCT
C87 .1
.75v
U24 LM317LM
L13 EXIC 4532U
C86 .1
C86 4.7/20
Vis 19V
R105 1.2k
RV9 500
R106 240
NOTES:
1) 27.3 5W ADDED AT POWER SUPPLY TO +8V
2) +19V REGULATOR ADDED AT POWER SUPPLY.
3) +8V ADJUSTED TO +7V AT POWER SUPPLY SO 19V REG. ADDED.
UNLESS OTHERWISE SPECIFIED
ALL RESISTORS 1/4 WATT 5%
ALL CAPACITORS IN uf
NOTES:
KR Research
Company KENSAL CORP.
Title REGULATORS
Sch REV
PCB REV B
Date 8/30/98
PCB NAME KL18238.PCB
SCH. NAME KL18238.SCH
SHEET 4 OF 4
Desc. BOECKELER KU-18283 RGB SLIDE SCANER

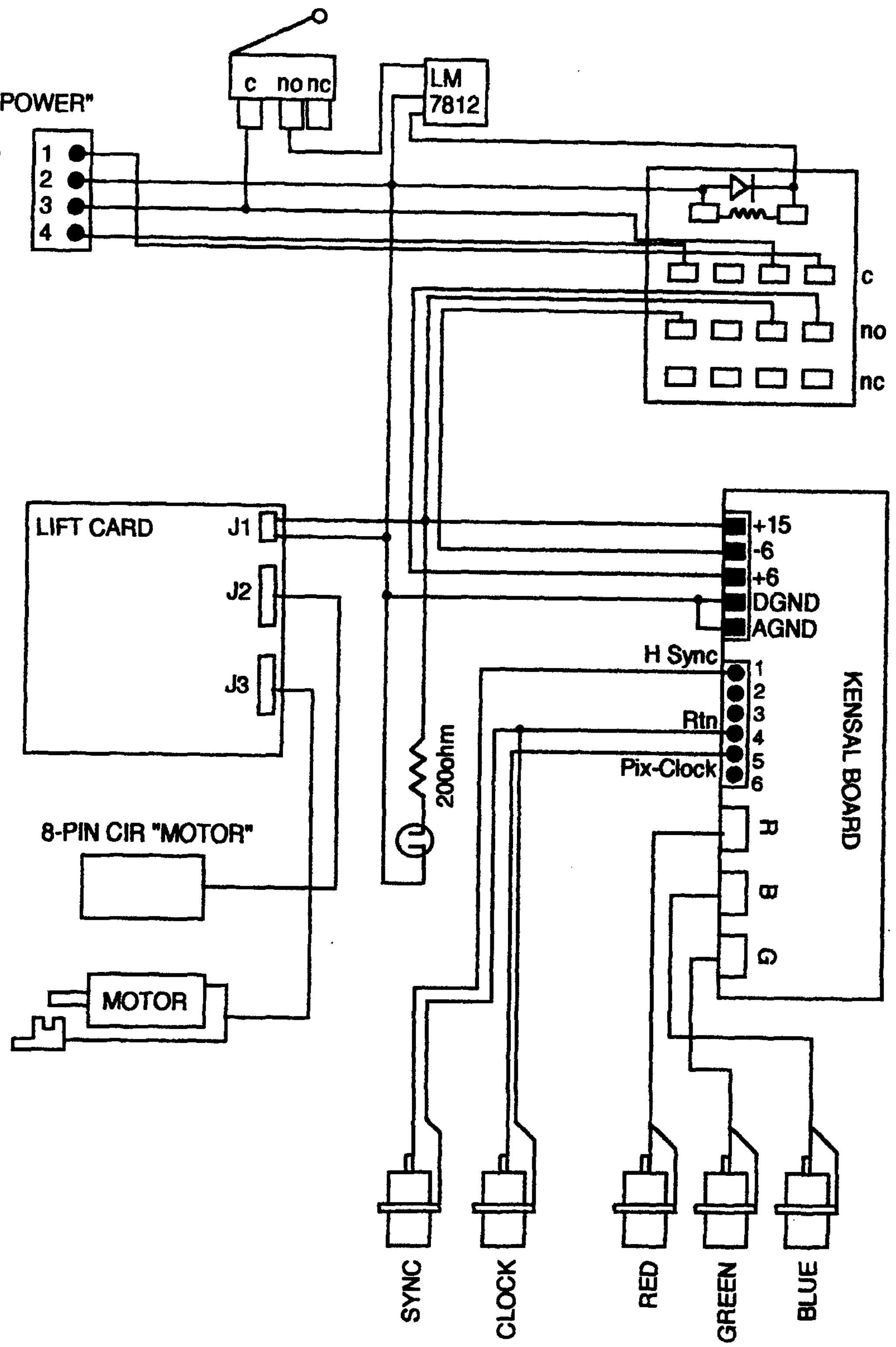
TRADE SECRETS
KENSAL CORP.



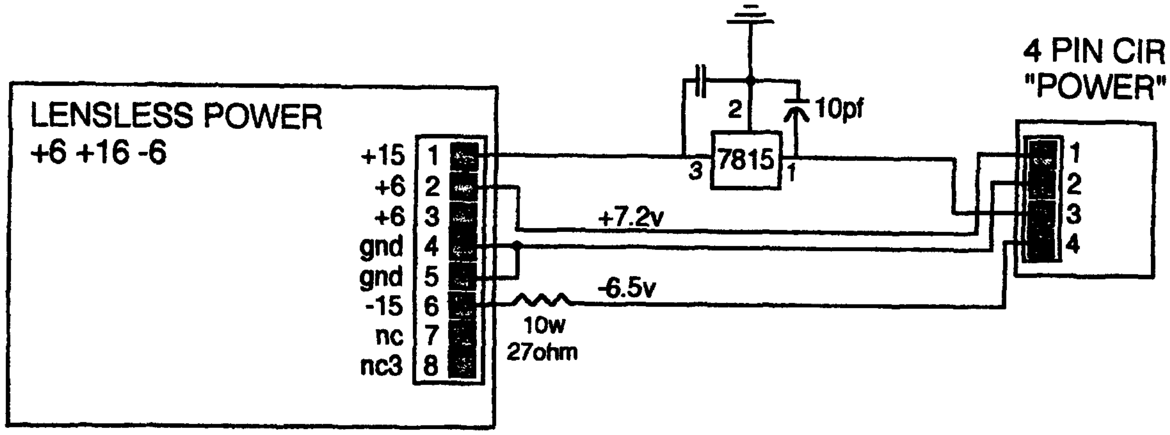

KENSAL CORP.
SAMPLE & HOLD
OUTPUT DRV.
BLUE OUT
GREEN OUT
RED OUT

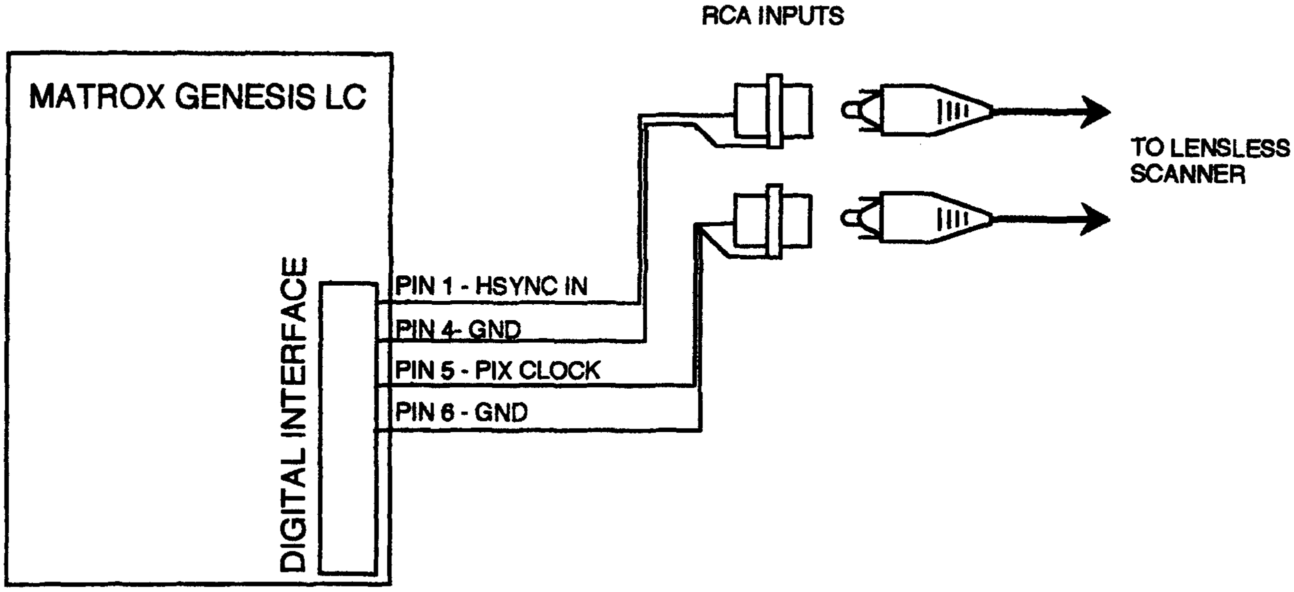
TRADE SECRETS
KENSAL CORP.





TSS LENSLESS SCANNER WIRE DIAGRAM



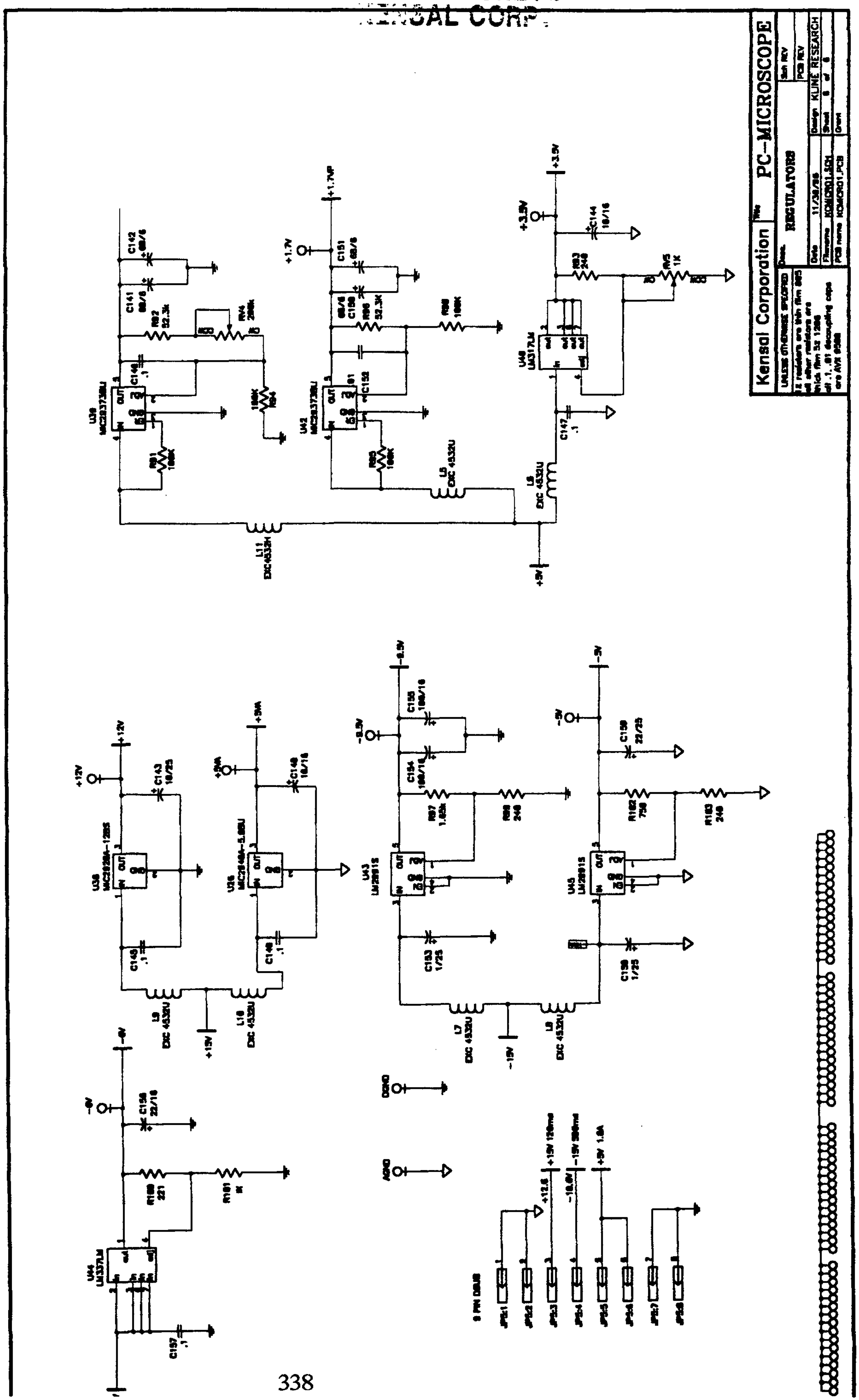


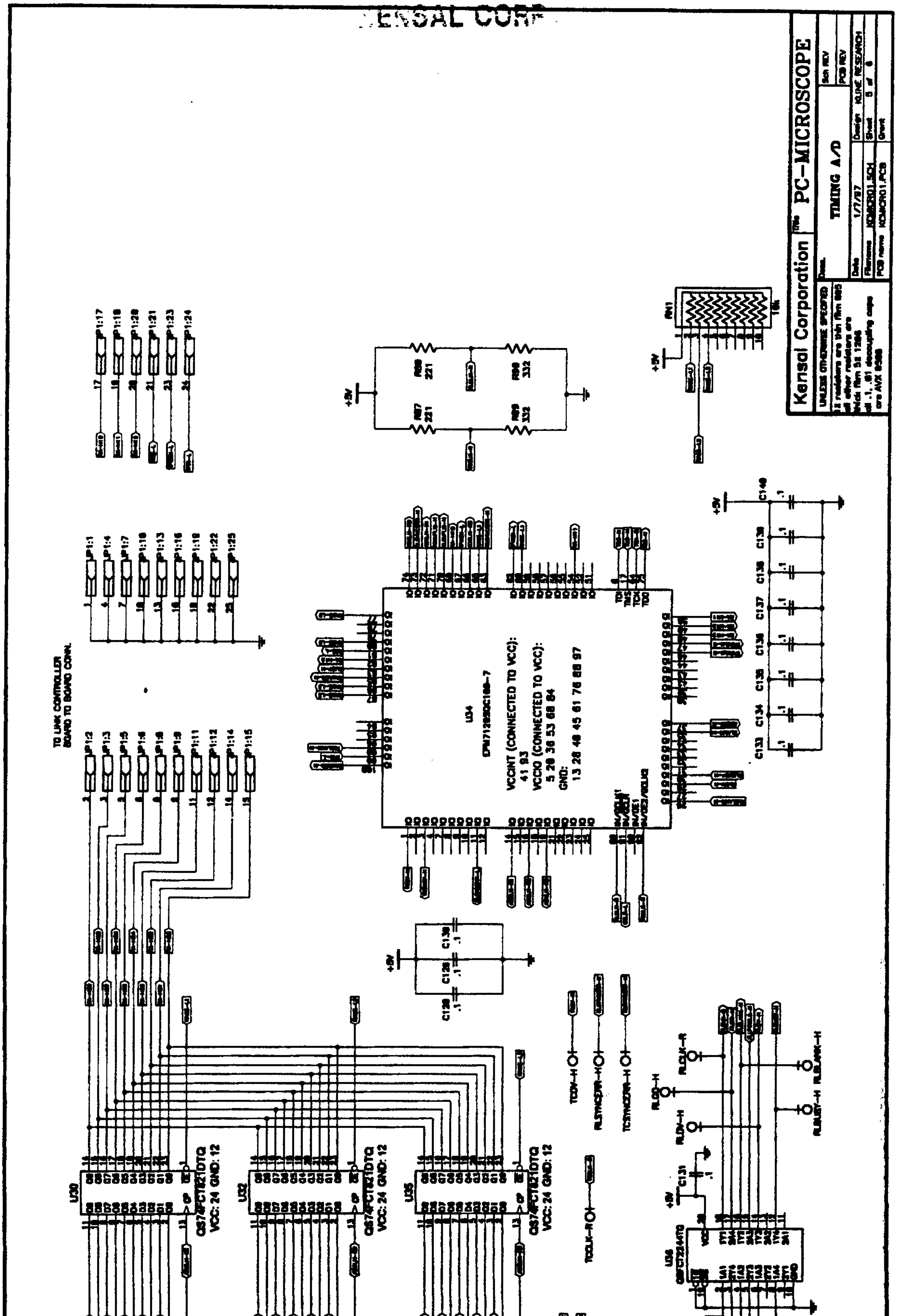
TSS POWER SUPPLY DIAGRAM

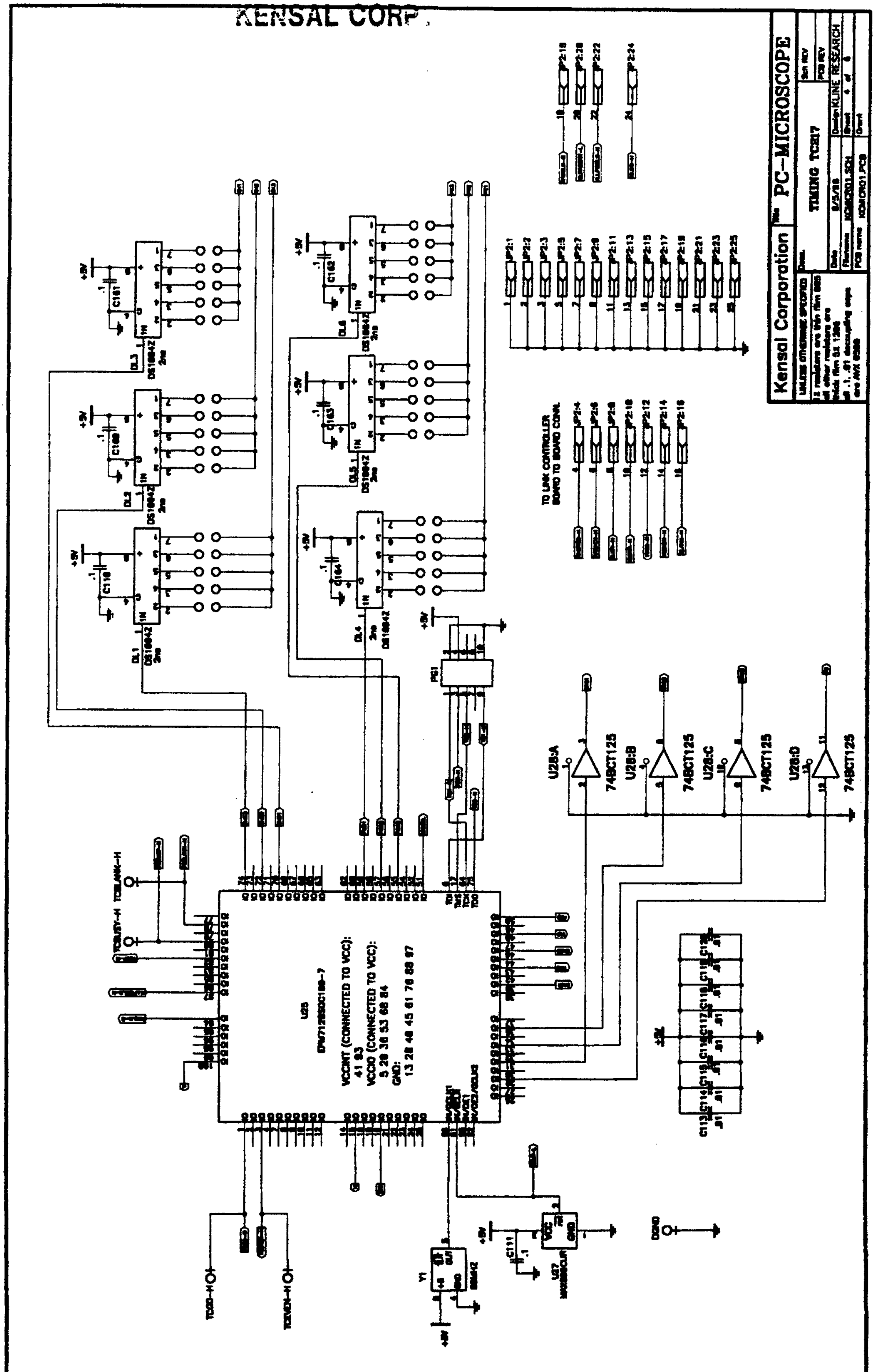
APPENDIX E

PCM SUPPORT DOCUMENTS

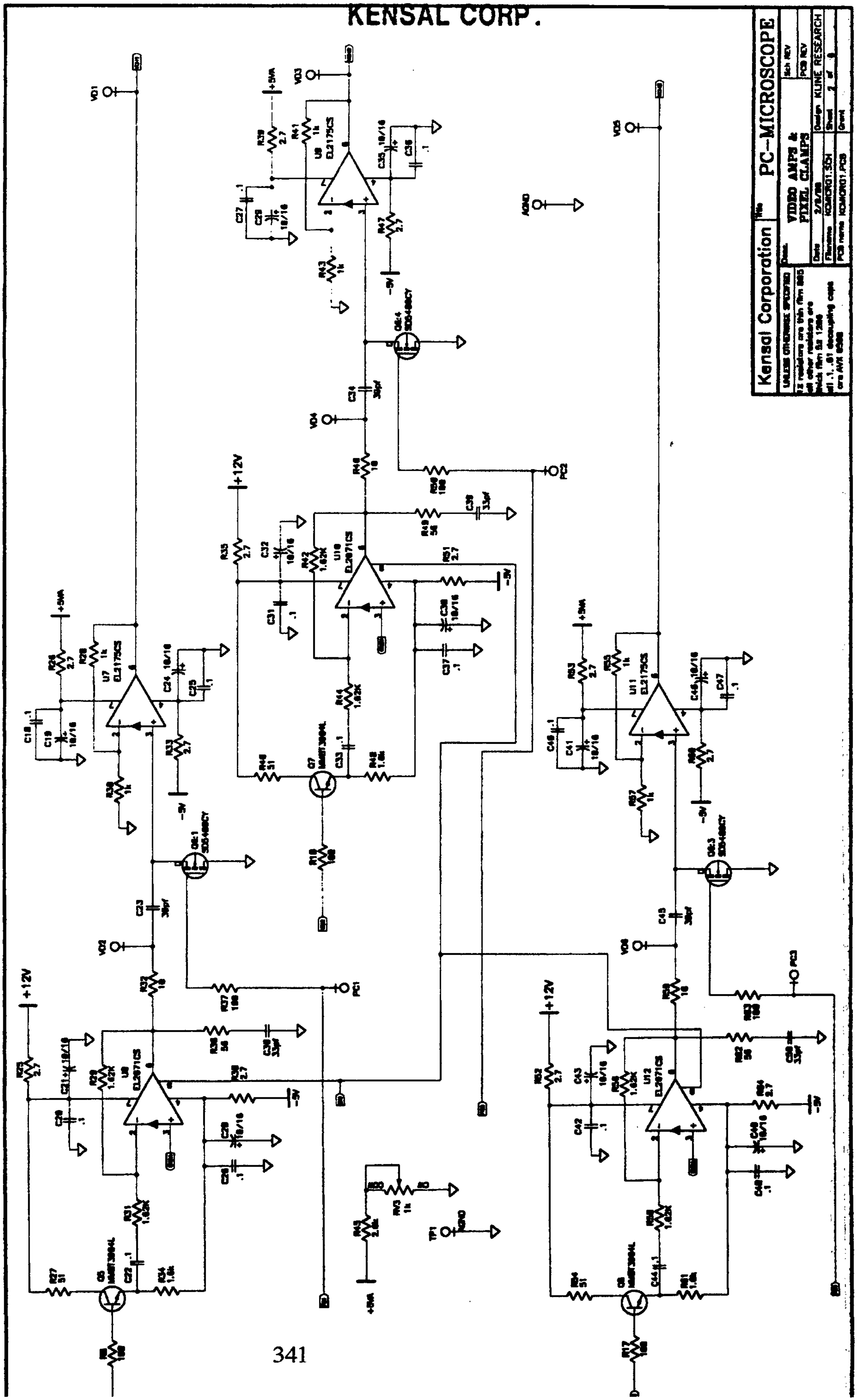
TRADE SECRETS
KENSAL CORP.



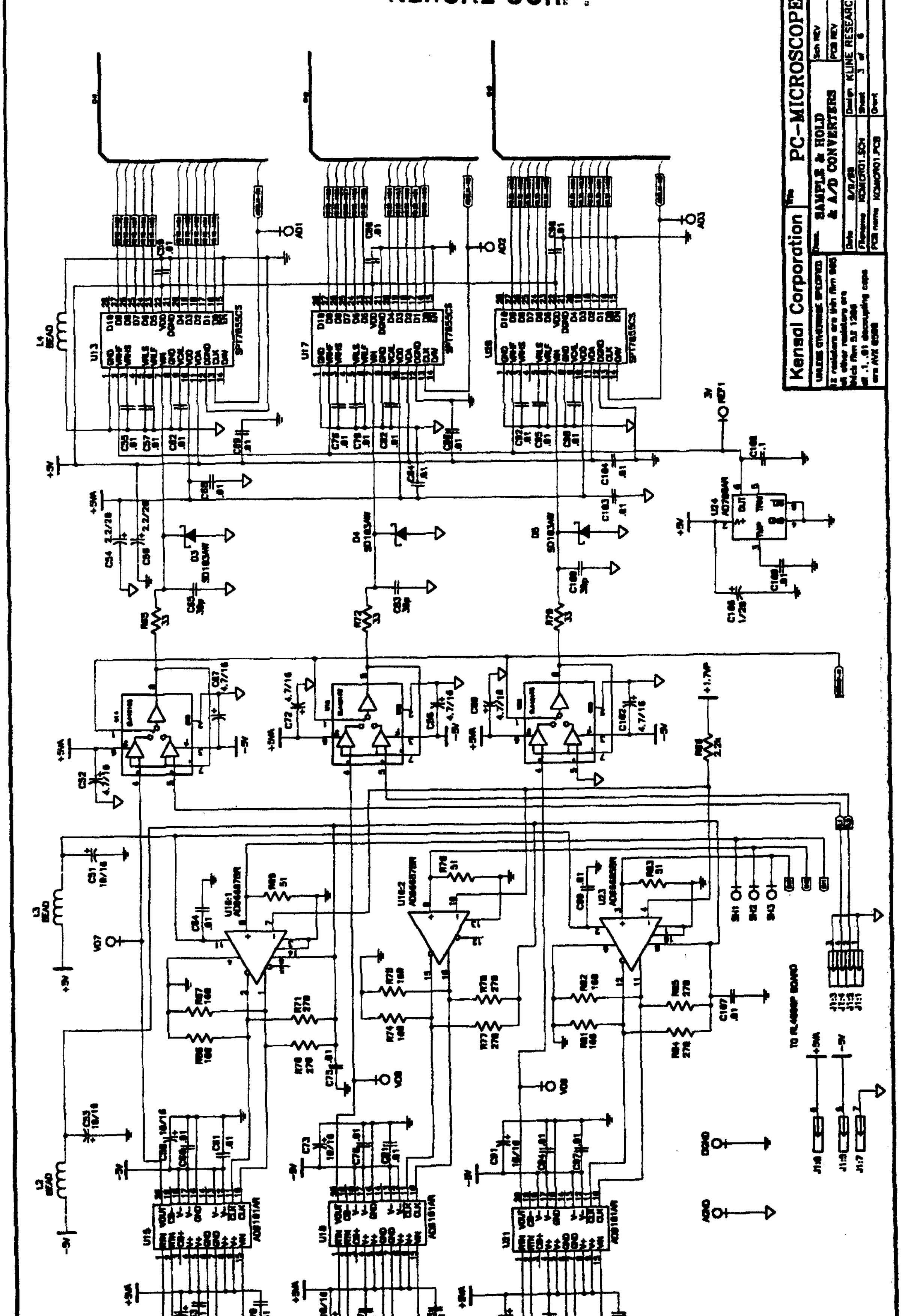

Kensal Corporation
PC-MICROSCOPE
TIMING A/D
1/7/97
KLINE RESEARCH
5 of 6
U34
U30
U32
U35
U36
QS74FCT821DTQ
VCC: 24 GND: 12
TO LINK CONTROLLER BOARD TO BOARD CONN.
VCCINT (CONNECTED TO VCC):
41 93
VCCIO (CONNECTED TO VCC):
5 20 36 53 68 84
GND:
13 28 48 45 61 76 88 97


Kensal Corporation
PC-MICROSCOPE

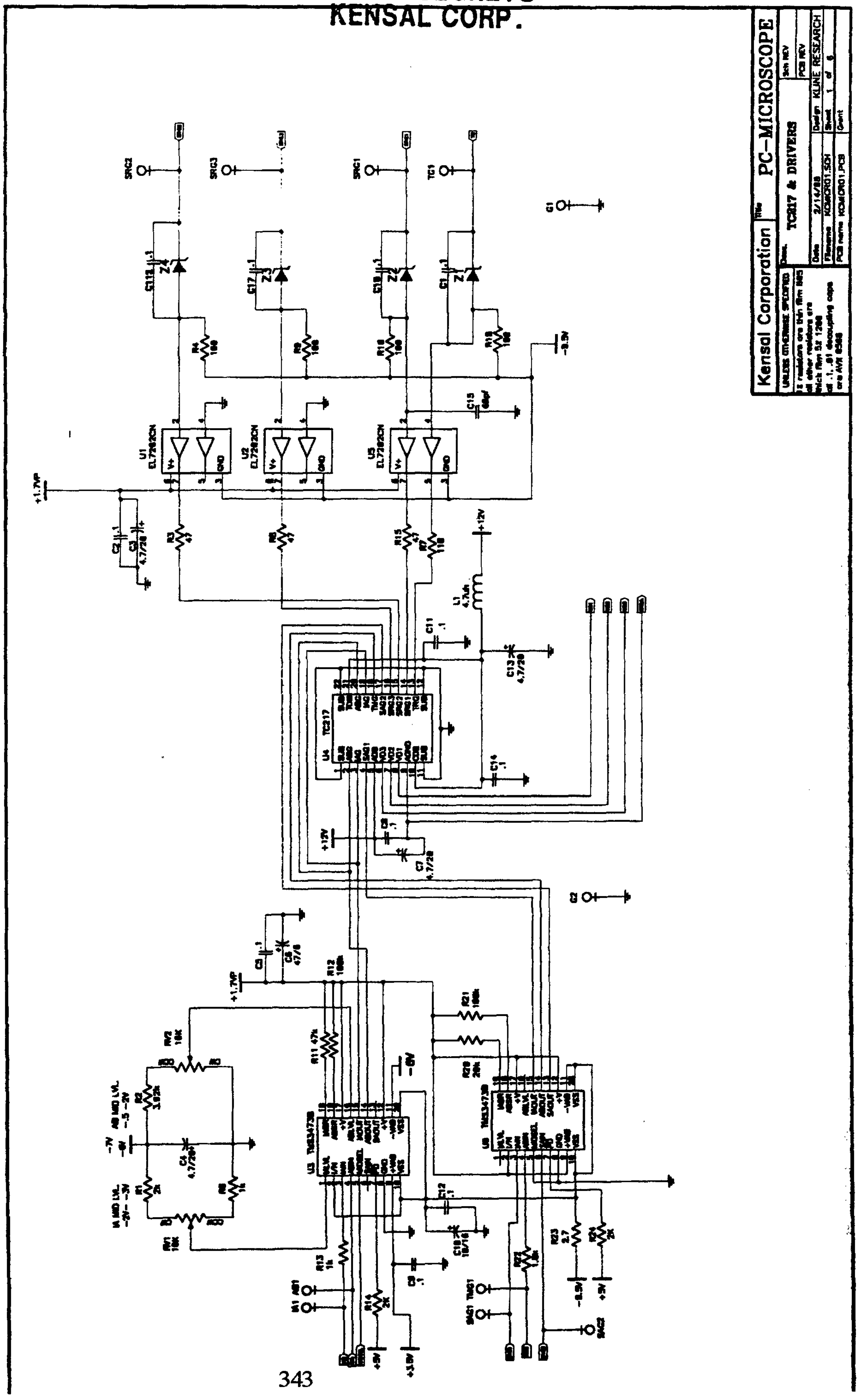
TRADE SECRETS
KENSAL CORP.



TRADE SECRETS
KENSAL CORP.



TRADE SECRETS
KENSAL CORP.



KENSAL CORPORATION TRADE SECRETS

Camera Head Register Definitions
Revision: D
Version Date: Monday, October 26, 1998

Summary

As mentioned in the document "High Level Design - Frame Capture/Link Controller", Rev. B (October 26, 1998, ~page 10), all data traffic between the Frame Capture Board and the Camera Head occurs in packets. In addition to the 256 possible data codes that can be transmitted in a data byte, approximately 11 command symbols (distinct from data bytes) can be transmitted and recognized. Of these, the Frame Capture Board/Camera Head uses 4, outlined below in the section "Packet Command Symbols". All packets consist of a header command symbol, a data selector byte, a variable number of data bytes and a trailer command symbol.

For example, lets say the user wanted to write to the CtrlStat register, then read it back to determine if the write succeeded. The Frame Capture Board would send the following word sequence to the Camera Head: WriteHdr, CtrlStat, <CtrlStat value>, EndPkt, ReadHdr, CtrlStat, EndPkt. The Camera Head would respond to the ReadHdr packet by sending the following back to the Frame Capture Board: DataHdr, CtrlStat, <CtrlStat value>, EndPkt.

Note that ReadHdr packets to "read" Even, Odd, or Line data are never sent from the Frame Capture Board to the Camera Head. The Camera Head automatically sends Even, Odd or Line data packets as a result of a write to the CtrlStat register requesting such data.

Packet Command Symbols ('tsc_h' active)

Value	Name	Direction	Description
0x02	WriteHdr	From PCI	Packet header to request write to Camera Head
0x03	ReadHdr	From PCI	Packet header to request read from Camera Head
0x04	DataHdr	From Cam	Packet header used for transmission of data from Camera Head
0x06	EndPkt	Both	Universal packet trailer

Data Descriptor Bytes ('tsc_h' inactive)

Value	Name	Direction	Description
0x00	CtrlStat	Both	Control/Status Register
0x01	ErrCnt	Both	Error Counter/Register
0x02	Delay	Both	Delay Registers
0x03	LED	Both	LED Intensity Registers
0x04	LUT	Both	Gain/Offset/Gamma LUT's
0xX5	Even	From Cam	Even TC217 Field
0xX6	Odd	From Cam	Odd TC217 Field
0xX7	Line	From Cam	RL4000P Linescan Frame

The 'X' for the Even, Odd and Line descriptors will be one of '1', '2', or '4' if the red, green or blue LED's were enabled during the exposure of the field. If 'rgbcycle_h' is set, then only one bit will be on at a time. If 'rgbcycle_h' isn't set, then all bits (ie. 0x7) will be on. Whether or not a particular LED was actually on is also determined by the contents of the respective LED Intensity register. If the register for a particular color was zero, then the LED, though enabled, was off during the exposure.

Control/Status Register

This register allows access to bits that control the camera head. See the section below regarding Control/Status Register bit definitions. The Camera Head initializes the CtrlStat register to 0x40 on powerup.

Error Counter/Register

This register contains a 6-bit counter that records the number of times the Hotlink receiver's RVS (receive violation symbol) signal went active since a zero was last written to this register. If the value in this register is non-zero, the red status LED on the Camera Head board will illuminate. The register also contains two other status bits (see below). The Camera Head initializes the ErrCnt register to 0x00 on powerup.

Delay Registers

These registers set the number of milliseconds to wait before initiating subsequent linescan or TC217 reads. After the Delay data descriptor byte is sent, three bytes representing the red, green, and blue delays are sent, followed by an EndPkt symbol. If the 'repeat_h' bit isn't set, the delay registers won't be used. If the 'rgbcycle_h' bit isn't set, then only the first register is used. If the 'rgbcycle_h' bit is set, all three registers are used in RGB sequence, depending on the particular color in use. The delay values may range from 0..255 milliseconds. Since neither of the sensors has shutters, any delay directly increases integration time of any light present. The Camera Head initializes the Delay registers to {0x00, 0x00, 0x00} on powerup.

LED Intensity Registers

These registers control the intensities of the red, green, and blue illuminator LEDs. After the LED data descriptor byte is sent, three bytes representing red, green, and blue intensities are sent, followed by an EndPkt symbol. Each intensity value may range from 0..255. A value of 0 means the corresponding LED is off. A value of 1 sets the corresponding LED's Pulse Width Modulator (PWM) to 10%. In other words, the LED will be on 10% of the time. A value of 255 sets the PWM to 80%. Values in between vary linearly from 10% to 80%. The Camera Head initializes the LED registers to {0x00, 0x00, 0x00} on powerup. If the 'repeat_h' and 'rgbcycle_h' bits are set, an automatic mode is entered wherein the LED colors are enabled one at a time in an RGB sequence. This mode assumes that all three registers will be loaded with non-zero values. For example, if the odd_h, even_h, repeat_h, and rgbcycle_h bits are set in the control register, the following frames will be exposed and read out without any program intervention: red odd, red even, green odd, green even, blue odd, blue even. The cycle will repeat indefinitely.

Gain/Offset/Gamma LUT's

As explained in the HLD document, there is a LookUp Table (LUT) within the Camera Head that maps 10-bit A/D converter data into the 8-bit data sent to the Frame Capture Board. The LUT is broken down into eight 1K pages as follows:

0x0000 - 0x03FF	TC217 Channel 1
0x0400 - 0x07FF	TC217 Channel 2
0x0800 - 0x0BFF	TC217 Channel 3
0x0C00 - 0x0FFF	Reserved (unused, should be 0x00)
0x1000 - 0x13FF	RL4000P Channel 1
0x1400 - 0x17FF	RL4000P Channel 2
0x1800 - 0x1BFF	Reserved (unused, should be 0x00)
0x1C00 - 0x1FFF	Reserved (unused, should be 0x00)

The LUT packet will always contain data for all LUT's (it is not possible to write/read one LUT only). For example, to write to the LUT's the following sequence is used: WriteHdr, LUT, <8192 bytes of LUT data>, EndPkt. The LUT's contain undefined values at powerup and require initialization from the Frame Capture Board.

Even TC217 Field

When an even TC217 field is requested, this data descriptor precedes the even field video data transferred to the Frame Capture PCB. This descriptor is never used in conjunction with a WriteHdr or ReadHdr symbol from the Frame Capture Board. If the 'allpixels_h' bit is set in the CtrlStat register, 488 lines of 1158 pixels (565,104 bytes) are transferred. The first two lines are dark. The makeup of the rest of the lines is: 1/2 dark pixel, 1134 active pixels, 1/2 dark and 22 fully dark pixels. If 'allpixels_h' is inactive, then 486 lines of 1140 (554,040) pixels will be transferred. The makeup of each line is: 1/2 dark pixel, 1134 active pixels, 1/2 dark and 4 fully dark pixels.

Odd TC217 Field

When an odd TC217 field is requested, this data descriptor precedes the odd field video data transferred to the Frame Capture PCB. The number of bytes transferred and the makeup of the video lines is identical to the even field.

RL4000P Linescan Frame

When an RL4000P linescan read is requested, this data descriptor precedes the video data transferred to the Frame Capture PCB. This descriptor is never used in conjunction with a WriteHdr or ReadHdr symbol from the Frame Capture Board. If the 'allpixels_h' bit is set in the CtrlStat register, 1 line of 4112 pixels (4112 bytes) are transferred. Of these, the first 16 are dark. If 'allpixels_h' is inactive, then 1 line of 4096 pixels (4096 bytes) will be transferred. All of these pixels are active.

CtrlStat Register Bit Definitions

Bit	Name	Write	Read
7	spare_h	Spare Bit	Spare Bit
6	rf_h	Hotlink Receiver Reframe	Hotlink Receiver Reframe
5	allpixels_h	Transmit All Pixels	Transmit All Pixels
4	rgbcycle_h	RGB Cycle	RGB Cycle
3	repeat_h	Repeat Selected Operation	Repeat Selected Operation
2	linescan_h	Read RL4000P Linescan Array	Read RL4000P Linescan Array
1	odd_h	Read TC217 Odd Field	Read TC217 Odd Field
0	even_h	Read TC217 Even Field	Read TC217 Even Field

even_h Read TC217 Even Field

If this bit is active, an even field readout will occur as soon as possible.
If both 'odd_h' and 'even_h' are set, the even field will be read out first, followed by the odd field. If 'repeat_h' is set, then the requested operation will repeat indefinitely, otherwise bit 'even_h' will be cleared after the readout.

odd_h Read TC217 Odd Field

If this bit is active, an odd field readout will occur as soon as possible.
If both 'odd_h' and 'even_h' are set, the even field will be read out first, followed by the odd field. If 'repeat_h' is set, then the requested operation will repeat indefinitely, otherwise bit 'odd_h' will be cleared after the readout.

line_h Read RL4000P Linescan Array

If this bit is active, a linescan array readout will occur as soon as possible.
If 'repeat_h' is set, then the requested operation will repeated indefinitely, otherwise bit 'linescan_h' will be cleared after the readout.

repeat_h Repeat Selected Operation

This bit works in conjunction with the 'even_h', 'odd_h' and 'linescan_h' bits to allow continuous readout of the requested sensor. The Delay Register is used to lengthen the exposure time between subsequent reads. The requested operation will repeat until the CtrlStat register is written with 'repeat_h' inactive.

rgbcycle_h RGB Cycle

This bit works in conjunction with the 'repeat_h' bit above to enable an automatic mode that sequentially exposes and reads the selected sensor with red, green and blue light, using the corresponding delays for each color without further program intervention.

allpixels_h Transmit All Pixels

For both the TC217 and RL4000P sensors, there are several pixels that are always dark. They can be used to determine what A/D converter value 'black' corresponds to. These pixels are digitized and sent to the Frame Capture Board along with the active video pixels when 'allpixels_h' is active. The downside to sending this data is that it will lengthen transmission time. For the fastest possible update, 'allpixels_h' should be '0'.

rf_h Hotlink Receiver Reframe

This bit is tied to the 'rf_h' signal of the Hotlink Receiver. Bit 'rf_h' should normally be set to '1', as this allows normal reframing to take place. Please refer to the Hotlink data sheet for a complete explanation of it's use.

spare_h Spare Bit

This bit is connected to a scope test point and a status LED on the Camera Head board. Otherwise it has no function.

ErrCnt Register Bit Definitions

Bit	Name	Write	Read
7	seqerr_h	0 -> clear bit, 1 -> NOP	Sequence Error
6	aderr_h	0 -> clear bit, 1 -> NOP	A/D Controller Error
5	errcnt_h[5]	Error Counter bit 5	Error Counter bit 5
4	errcnt_h[4]	Error Counter bit 4	Error Counter bit 4
3	errcnt_h[3]	Error Counter bit 3	Error Counter bit 3
2	errcnt_h[2]	Error Counter bit 2	Error Counter bit 2
1	errcnt_h[1]	Error Counter bit 1	Error Counter bit 1
0	errcnt_h[0]	Error Counter bit 0	Error Counter bit 0

seqerr_h Sequence Error

If this bit is active, an unexpected character or symbol was received during packet decoding. The following 6 conditions would cause this error:

1) A data character was received when a packet header was expected.
2) A command symbol was received when a data descriptor byte was expected.
3) A command symbol was received while writing data to the LUT RAM.
4) A command symbol was received while writing data to a register.
5) A command or data character other than EndPkt was received when an EndPkt was expected.
6) A command or data character was received while the Camera Head was processing a response to a ReadHdr packet.

This bit is cleared by writing a '0' into it. Writing a '1' has no effect.

aderr_h A/D Controller Error

If this bit is active, the A/D Controller has detected one or more of the following 4 conditions:

1) There was a clock synchronization error between the TC217 Controller and the A/D Controller.
2) There was a clock synchronization error between the RL4000P Controller and the A/D Controller.
3) The transmit FIFO overflowed while processing TC217 data.
4) The transmit FIFO overflowed while processing RL4000P data.

This bit is cleared by writing a '0' into it. Writing a '1' has no effect.

errcnt_h[5..0] Error Counter

This 6-bit counter records the number of times the RVS (receive violation symbol) bit on the Hotlink receiver pulsed. If the counter gets to 0x3F it will stop counting (ie. it won't roll over to 0x00). If this register is non-zero the red "Error" LED will light. This register is read/write.

```
%===========================================================================
LED Pulse Width Modulator

Kensal Corporation
OSC Project

This module defines one channel of pulse width modulation. Constants MINOFF
and MINON set the maximum and minimum duty cycles, respectively, when the
PWM is active. If ratio_h[] is zero, the PWM is disabled and the LED is
off. If ratio_h[] is '1', then the LED will be on MINON + 3 cycles and off
255 plus MINOFF cycles. If ratio_h[] is '255', then the LED will be on MINON
plus 257 cycles and off MINOFF plus 1 cycles.

In summary, for non-zero values of ratio_h[] the percentage on-time is defined as:
        100 * (MINON + ratio_h[] + 2) / (MINON + MINOFF + 258)

Unfortunately, the two variables MINOFF and MINON interrelate when correlating
to the minimum and maximum desired duty cycle. The Mac Excel spreadsheet
titled "LED PWM Calculation" makes short work of solving for MINOFF and MINON
given the desired minimum and maximum duty cycles. The values used below will
produce a minimum duty cycle of 10 percent and a maximum of 80 percent.

Author: Ken Crocker
Date:   13 Sep 96
File:   LEDPWM.TDF
Rev:    1.0
===========================================================================%
TITLE "LED Pulse Width Modulator";

CONSTANT MINOFF     = 71;              -- minimum off time
CONSTANT MINON      = 33;              -- minimum on time
CONSTANT CNTRCLR    = MINOFF + MINON + 1;

SUBDESIGN ledpwm
(
    clk_r           : INPUT;        -- 33 MHz system clock
    clr_l           : INPUT;        -- active low asynchronous system reset

    enable_h        : INPUT;        -- overrides ratio_h[]
    ratio_h[7..0]   : INPUT;        -- can range from 0..255, 0 -> LED off
    on_h            : OUTPUT;
)

VARIABLE
    enabled_h       : LCELL;

    cnt_h[7..0]     : DFF;
    cntclr_h        : LCELL;
    cnttcoffdelay_h : LCELL;
    cnttcondelay_h  : LCELL;
    cnttcon_h       : LCELL;
    cnttcoff_h      : LCELL;

    pwSM            : MACHINE OF BITS (
                        on_h
                    ) WITH STATES (
                        pwIdle      = B"0",
                        pwOffDelay  = B"0",
                        pwOnDelay   = B"1",
                        pwOn        = B"1",
                        pwOff       = B"0"
                    );

BEGIN
```

```
    enabled_h   = enable_h & (ratio_h[] != 0);              -- LCELL

    cnt_h[].clk      = clk_r;
    !cnt_h[].clrn    = !clr_l;
    IF cntclr_h THEN
        cnt_h[].d = H"00";
    ELSE
        cnt_h[].d = cnt_h[].q + 1;
    END IF;

    cnttcoffdelay_h = cnt_h[] == MINOFF;                     -- LCELL
    cnttcondelay_h  = cnt_h[] == CNTRCLR;                    -- LCELL
    cnttcon_h       = cnt_h[] == ratio_h[];                  -- LCELL
    cnttcoff_h      = cnt_h[] == H"FF";                      -- LCELL

    cntclr_h        = pwOnDelay & (cnt_h[] == CNTRCLR);      -- LCELL

    pwSM.clk        = clk_r;
    pwSM.reset      = !clr_l;

    CASE pwSM IS
        WHEN pwIdle =>          -- <nothing> active
            cntclr_h = VCC;
            IF enabled_h THEN pwSM = pwOffDelay;
            END IF;

        WHEN pwOffDelay =>      -- <nothing> active
            IF !enabled_h THEN pwSM = pwIdle;
            ELSIF cnttcoffdelay_h THEN pwSM = pwOnDelay;
            END IF;

        WHEN pwOnDelay =>       -- on_h active
            IF !enabled_h THEN pwSM = pwIdle;
            ELSIF cnttcondelay_h THEN pwSM = pwOn;
            END IF;

        WHEN pwOn =>            -- on_h active
            IF !enabled_h THEN pwSM = pwIdle;
            ELSIF cnttcoff_h THEN
                IF enabled_h THEN pwSM = pwOffDelay;
                ELSE pwSM = pwIdle;
                END IF;
            ELSIF cnttcon_h THEN pwSM = pwOff;
            END IF;

        WHEN pwOff =>           -- <nothing> active
            IF !enabled_h THEN pwSM = pwIdle;
            ELSIF cnttcoff_h THEN
                IF enabled_h THEN pwSM = pwOffDelay;
                ELSE pwSM = pwIdle;
                END IF;
            END IF;
    END CASE;
END;
```

```
%=========================================================================
Hotlink Controller

Kensal Corporation
OSC Project

Author: Ken Crocker
Date:   27 Sep 96
File:   LINK.TDF
Rev:    1.0
=========================================================================%
TITLE "Hotlink Controller";

INCLUDE "LEDPWM.INC";
INCLUDE "STRETCH.INC";

-- Packet Command Symbols ('rsc_h' active)
CONSTANT WriteHdr    = H"02";
CONSTANT ReadHdr     = H"03";
CONSTANT DataHdr     = H"04";
CONSTANT Sync        = H"05";
CONSTANT EndPkt      = H"06";

-- Data Descriptor Bytes ('rsc_h' inactive)
CONSTANT CtrlStat    = H"00";
CONSTANT ErrCnt      = H"01";
CONSTANT Delay       = H"02";
CONSTANT LED         = H"03";
CONSTANT LUT         = H"04";
CONSTANT Even        = H"05";    -- partial: LED value or'd into MS 4-bits
CONSTANT Odd         = H"06";    -- partial: LED value or'd into MS 4-bits
CONSTANT Line        = H"07";    -- partial: LED value or'd into MS 4-bits

-- Data Descriptor LED Nybbles
CONSTANT LEDRed      = H"10";
CONSTANT LEDGrn      = H"20";
CONSTANT LEDBlu      = H"40";

-- Enum Values for seq_h[]
CONSTANT tDataHdr    = 0;
CONSTANT tDataDesc   = 1;
CONSTANT tData       = 2;
CONSTANT tEndPkt     = 3;

SUBDESIGN link (
    gclk33_r            : INPUT;        -- 33 MHz system clock
    gckr_r              : INPUT;        -- Receive Hotlink Clock
    gclr_l              : INPUT;        -- active low asynchronous system reset

    -- Hotlink Receiver Interface
    rd_h[7..0]          : INPUT;
    rsc_h               : INPUT;
    rvs_h               : INPUT;
    rdy_l               : INPUT;
    rbisten_l           : OUTPUT;       -- (slow slew rate)
    rf_h                : OUTPUT;       -- (slow slew rate)
    rselb_l             : OUTPUT;       -- (slow slew rate)

    -- Hotlink Transmitter Interface
    td_h[7..0]          : OUTPUT;
    tsc_h               : OUTPUT;
    tbisten_l           : OUTPUT;       -- (slow slew rate)
    tenn_l              : OUTPUT;
```

```
    -- Transmit FIFO Interface
    tfor_l              : INPUT;
    entfren_l           : OUTPUT;
    tena_l              : INPUT;
    tfprs_l             : OUTPUT;       -- (slow slew rate)
    tffs_h              : OUTPUT;       -- (slow slew rate)

    -- TC217 Controller Interface
    tcgo_h              : OUTPUT;
    tceven_h            : OUTPUT;

    -- RL4000P Controller Interface
    rlgo_h              : OUTPUT;

    -- Signals Common to TC217 and RL4000P
    allpixels_h         : OUTPUT;       -- (slow slew rate)

    -- A/D Controller Interface
    ramreq_h            : OUTPUT;       -- (slow slew rate)
    ramack_h            : INPUT;
    tcclksel_h          : INPUT;
    busy_h              : INPUT;
    blank_h             : INPUT;
    aderr_h             : INPUT;
    clraderr_l          : OUTPUT;       -- (slow slew rate)

    -- LUT RAM Interface Signals
    ra_h[12..00]        : OUTPUT;       -- (slow slew rate)
    rdat_h[7..0]        : BIDIR;        -- (slow slew rate)
    rwe_l               : OUTPUT;       -- (slow slew rate)
    roe_l               : OUTPUT;       -- (slow slew rate)

    -- LED Microscope Slide Illuminator Outputs
    redon_h             : OUTPUT;
    grnon_h             : OUTPUT;
    bluon_h             : OUTPUT;

    -- Diagnostic Signals
    error_l             : OUTPUT;       -- active if errcnt_h[] is non-zero (red LED)(slow slew rate)
    spare_l             : OUTPUT;       -- inverted from of 'spare_h' (yel LED)(slow slew rate)
    rbusy_l             : OUTPUT;       -- pulse stretched version of 'rdy_l' (grn LED)
    tbusy_l             : OUTPUT;
    bistin_h            : INPUT;
    bistout_h           : OUTPUT;       -- used for positive feedback (debouncing)(slow slew rate)
    bypassin_h          : INPUT;
    bypassout_h         : OUTPUT;       -- used for positive feedback (debouncing)(slow slew rate)

    oops_l              : INPUT;        -- correction for miswiring on PCB, remove on next rev!
)

VARIABLE
    clk_r               : NODE;         -- 33 MHz crystal oscillator clock
    ckr_r               : NODE;
    clr_l               : NODE;
    jmpclr_l            : DFF;

    -- Receive Hotlink Signals
    rdin_h[7..0]        : DFF;          -- F/F is in IOCELL
    rscin_h             : DFF;          -- F/F is in IOCELL
    rvsin_h             : DFF;          -- F/F is in IOCELL
    rdyin_l             : DFF;          -- F/F is in IOCELL

    -- Transmit Hotlink Signals
    regout_h[7..0]      : LCELL;
```

```
tdout_h[7..0]         : DFF;
tdtri_h[7..0]         : TRI;
tscout_h              : DFF;
tbisten_l             : SRFF;

-- A/D Controller Signals
busysync_h            : DFF;           -- F/F is in IOCELL
blanksync_h           : DFF;           -- F/F is in IOCELL
aderrin_h             : DFF;           -- F/F is in IOCELL
clraderr_l            : SRFF;

rdesc_h[7..0]         : DFF;           -- receive data descriptor byte
tdesc_h[7..0]         : DFF;           -- transmit data descriptor byte

redled                : ledpwm;
grnled                : ledpwm;
bluled                : ledpwm;

enred_h               : NODE;
engrn_h               : NODE;
enblu_h               : NODE;
color_h[2..0]         : DFF;
colorclr_h            : DFF;

ctrlstatwe_h          : LCELL;
even_h                : DFF;
odd_h                 : DFF;
line_h                : DFF;
repeat_h              : DFFE;
rgbcycle_h            : DFFE;
allpixels_h           : DFFE;
rf_h                  : DFFE;
spare_l               : DFFE;

clreven_h             : NODE;
clrodd_h              : NODE;
clrline_h             : NODE;

errcnt_h[5..0]        : DFF;
errcntwe_h            : LCELL;
errcntinc_h           : LCELL;

seqerr_h              : SRFF;
setseqerr_h           : LCELL;

redreg_h[7..0]        : DFFE;
redregwe_h            : LCELL;
grnreg_h[7..0]        : DFFE;
grnregwe_h            : LCELL;
blureg_h[7..0]        : DFFE;
bluregwe_h            : LCELL;

reddelay_h[7..0]      : DFFE;
reddelaywe_h          : LCELL;
grndelay_h[7..0]      : DFFE;
grndelaywe_h          : LCELL;
bludelay_h[7..0]      : DFFE;
bludelaywe_h          : LCELL;
delayvalue_h[7..0]    : LCELL;
delay_h[7..0]         : DFF;
delaytc_h             : LCELL;
msdelay_h[15..00]     : DFF;
msdelayld_h           : NODE;
msdelaytc_h           : DFF;
```

```
msdelaytc1_h        : LCELL;
borrow_h            : LCELL;

dlSM                : MACHINE OF BITS (
                        delayld_h, delaycnt_h, delaydone_h
                      ) WITH STATES (
                        dl00 = B"100",
                        dl01 = B"010",
                        dl02 = B"000",
                        dl03 = B"001"
                      );

rra_h[12..00]       : DFF;
rratc_h             : LCELL;
tra_h[12..00]       : DFF;
tratc_h             : LCELL;
ratc_h[12..00]      : NODE;
raout_h[12..00]     : NODE;
ratri_h[12..00]     : TRI;
rdatoutld_h         : NODE;
rdatout_h[7..0]     : DFFE;      -- register is in IOCELL
rdattri_h[7..0]     : TRI;
raoe_l              : DFF;
rdatreq_h           : DFF;
rwe_h               : NODE;
ramreq_h            : SRFF;

rcSM                : MACHINE OF BITS (
                        roe_l, rdatoe_l
                      ) WITH STATES (
                        rc00    = B"11",
                        rc01    = B"10",
                        rc02    = B"01"
                      );

oneshot1            : stretch;
oneshot2            : stretch;

rhbistsync_h        : DFF;
rdacksync_h         : DFF;
rrainc_h            : LCELL;
we_h                : NODE;
syncbyte_h          : NODE;      -- was LCELL
rhSM                : MACHINE OF BITS (
                        rdescen_h, qread_h, noerrcnt_h, bisten_h
                      ) WITH STATES (
                        rhIdle       = B"0000",
                        rhWrite      = B"1000",
                        rhLUTData    = B"0000",
                        rhWrLUT      = B"0000",
                        rhRegData    = B"0000",
                        rhWrEndPkt   = B"0000",
                        rhRead       = B"1000",
                        rhRdEndPkt   = B"0000",
                        rhQRead      = B"0100",
                        rhBIST1      = B"0010",
                        rhBIST2      = B"0011",
                        rhBIST3      = B"0001",
                        rhBIST4      = B"0011",
                        rhBIST5      = B"0010"
                      );

ctrlstatwed_h       : DFF;
clrsm_l             : LCELL;
```

```
    qreadsync_h         : DFF;
    video_h             : DFFE;
    qbistsync_h         : DFF;
    lutsel_h            : LCELL;
    thSM                : MACHINE OF BITS (
                            seq_h[1..0], traclr_h, trainc_h, tenn_l, tdtrioe_l, entfren_l, rdack_h
                        ) WITH STATES (
                            thIdle       = B"01001010",
                            thDatHdr     = B"00100010",
                            thDatDesc    = B"01000010",
                            thRdLUT      = B"10001010",
                            thDoRd       = B"10010010",
                            thEndRd      = B"11001011",
                            thSndEndPkt  = B"11000010",
                            thDoVid      = B"01001010",
                            thVidDat1    = B"01001110",
                            thVidDat2    = B"01001100",
                            thVidDat3    = B"01001110",
                            thBIST       = B"01000010"
                        );

    evensync_h          : DFF;
    oddsync_h           : DFF;
    linesync_h          : DFF;
    chSM                : MACHINE OF BITS (
                            tceven_h, tcgo_h, rlgo_h,
                            senddesc_h, sendtrailer_h,
                            dodelay_h, nextcolor_h
                        ) WITH STATES (
                            chIdle       = B"0000000",
                            chEvenDelay  = B"0000010",
                            chEven1      = B"1100000",
                            chEven2      = B"1101000",
                            chEven3      = B"0000000",
                            chTrailer    = B"0000100",
                            chOddDelay   = B"0000010",
                            chOdd1       = B"0100000",
                            chOdd2       = B"0101000",
                            chOdd3       = B"0000000",
                            chLineDelay  = B"0000010",
                            chLine1      = B"0010000",
                            chLine2      = B"0011000",
                            chLine3      = B"0000000",
                            chColor1     = B"0000101",
                            chColor2     = B"0000100"
                        );

    tcclkselin_h        : DFF;      -- register is in IOCELL
    tfSM                : MACHINE OF BITS (
                            tfprs_l, tffs_h
                        ) WITH STATES (
                            tfAlmostRL   = B"00",
                            tfRL         = B"10",
                            tfAlmostTC   = B"01",
                            tfTC         = B"11"
                        );

BEGIN
    -- Global Clock and Asynchronous Clear Signals
    clk_r           = GLOBAL (gclk33_r);    -- 33.333 MHz clock oscillator
    ckr_r           = GLOBAL (gckr_r);      -- 33 MHz Receive Hotlink PLL Clock
    clr_l           = GLOBAL (gclr_l);

    -- CtrlStat Register
```

```
    --  rd_h[7] -> spare_h
    --  rd_h[6] -> rf_h
    --  rd_h[5] -> allpixels_h
    --  rd_h[4] -> rgbcycle_h
    --  rd_h[3] -> repeat_h
    --  rd_h[2] -> line_h
    --  rd_h[1] -> odd_h
    --  rd_h[0] -> even_h
    even_h.clk      = ckr_r;
    !even_h.prn     = !clr_l;              -- powerup w/even_h active!
--  !even_h.clrn    = !clr_l;
    IF clreven_h THEN
        even_h.d = GND;
    ELSIF ctrlstatwe_h THEN
        even_h.d = rdin_h[0];
    ELSE
        even_h.d = even_h.q;
    END IF;

    odd_h.clk       = ckr_r;
    !odd_h.clrn     = !clr_l;
    IF clrodd_h THEN
        odd_h.d = GND;
    ELSIF ctrlstatwe_h THEN
        odd_h.d = rdin_h[1];
    ELSE
        odd_h.d = odd_h.q;
    END IF;

    line_h.clk      = ckr_r;
    !line_h.clrn    = !clr_l;
    IF clrline_h THEN
        line_h.d = GND;
    ELSIF ctrlstatwe_h THEN
        line_h.d = rdin_h[2];
    ELSE
        line_h.d = line_h.q;
    END IF;

    repeat_h.clk         = ckr_r;
    !repeat_h.prn        = !clr_l;
--  !repeat_h.clrn       = !clr_l;          -- powerup w/repeat_h active!
    repeat_h.ena         = ctrlstatwe_h;
    repeat_h.d           = rdin_h[3];

    rgbcycle_h.clk       = ckr_r;
    !rgbcycle_h.prn      = !clr_l;          -- powerup w/rgbcycle_h active!
--  !rgbcycle_h.clrn     = !clr_l;
    rgbcycle_h.ena       = ctrlstatwe_h;
    rgbcycle_h.d         = rdin_h[4];

    allpixels_h.clk      = ckr_r;
    !allpixels_h.clrn    = !clr_l;
    allpixels_h.ena      = ctrlstatwe_h;
    allpixels_h.d        = rdin_h[5];

    rf_h.clk             = ckr_r;
    !rf_h.prn            = !clr_l;
    rf_h.ena             = ctrlstatwe_h;
    rf_h.d               = rdin_h[6];

    spare_l.clk          = ckr_r;
    !spare_l.prn         = !clr_l;
    spare_l.ena          = ctrlstatwe_h;
```

```
!spare_l.d          = rdin_h[7];

-- ErrCnt Counter/Register
-- Sequence Error Status Bit
seqerr_h.clk        = ckr_r;
!seqerr_h.clrn      = !clr_l;
seqerr_h.r          = seqerr_h.q & errcntwe_h & !rdin_h[7] & !setseqerr_h;
seqerr_h.s          = setseqerr_h;

-- A/D Error Status Bit
aderrin_h.clk       = ckr_r;
!aderrin_h.clrn     = !clr_l;
aderrin_h.d         = aderr_h;

clraderr_l.clk      = ckr_r;
!clraderr_l.prn     = !clr_l;
clraderr_l.s        = !aderrin_h;
clraderr_l.r        = aderrin_h & errcntwe_h & !rdin_h[6];

-- Error Counter
errcnt_h[].clk      = ckr_r;
!errcnt_h[].clrn    = !clr_l;
IF errcntwe_h THEN
    errcnt_h[].d = rdin_h[5..0];
ELSIF errcntinc_h THEN
    errcnt_h[].d = errcnt_h[].q + 1;
ELSE
    errcnt_h[].d = errcnt_h[].q;
END IF;
errcntinc_h     = !noerrcnt_h & !rdyin_l & rvsin_h & (errcnt_h[] != 63);    -- LCELL

-- Delay Registers
reddelay_h[].clk    = ckr_r;
!reddelay_h[].clrn  = !clr_l;
reddelay_h[].ena    = reddelaywe_h;
reddelay_h[].d      = rdin_h[];

grndelay_h[].clk    = ckr_r;
!grndelay_h[].clrn  = !clr_l;
grndelay_h[].ena    = grndelaywe_h;
grndelay_h[].d      = rdin_h[];

bludelay_h[].clk    = ckr_r;
!bludelay_h[].clrn  = !clr_l;
bludelay_h[].ena    = bludelaywe_h;
bludelay_h[].d      = rdin_h[];

-- LED Registers
redreg_h[].clk      = ckr_r;
!redreg_h[].clrn    = !clr_l;
redreg_h[].ena      = redregwe_h;
redreg_h[].d        = rdin_h[];

grnreg_h[].clk      = ckr_r;
!grnreg_h[].clrn    = !clr_l;
grnreg_h[].ena      = grnregwe_h;
grnreg_h[].d        = rdin_h[];

blureg_h[].clk      = ckr_r;
!blureg_h[].clrn    = !clr_l;
blureg_h[].ena      = bluregwe_h;
blureg_h[].d        = rdin_h[];

-- Receive Channel Register/LUT Address Counter
```

```
rra_h[].clk      = ckr_r;
!rra_h[].clrn    = !clr_l;
IF rdescen_h THEN
    rra_h[].d = 0;
ELSIF rrainc_h THEN
    rra_h[].d = rra_h[].q + 1;
ELSE
    rra_h[].d = rra_h[].q;
END IF;
rratc_h = (rra_h[] == ratc_h[]);     -- LCELL

-- Transmit Channel Register/LUT Address Counter
tra_h[].clk      = clk_r;
!tra_h[].clrn    = !clr_l;
IF traclr_h THEN
    tra_h[].d = 0;
ELSIF trainc_h THEN
    tra_h[].d = tra_h[].q + 1;
ELSE
    tra_h[].d = tra_h[].q;
END IF;
tratc_h = (tra_h[] == ratc_h[]);     -- LCELL

-- Register/LUT Address MUX
IF qread_h THEN
    raout_h[] = tra_h[].q;
ELSE
    raout_h[] = rra_h[].q;
END IF;
ratri_h[].in     = raout_h[];
ratri_h[].oe     = !raoe_l;
ra_h[]           = ratri_h[].out;

-- RAM Control Signals
ramreq_h.clk     = ckr_r;
!ramreq_h.clrn   = !clr_l;
ramreq_h.s       = rdescen_h & !rdyin_l & !rscin_h & (rdin_h[] == LUT);
ramreq_h.r       = (rhSM == rhIdle);

raoe_l.clk       = ckr_r;
!raoe_l.d        = ramack_h;

!rwe_l           =  rwe_h;

-- RAM Data I/O Signals
rdatout_h[].clk      = ckr_r;
rdatout_h[].ena      = rdatoutld_h;
rdatout_h[].d        = rdin_h[].q;
rdattri_h[].in       = rdatout_h[].q;
rdattri_h[].oe       = !rdatoe_l;
rdat_h[]             = rdattri_h[].out;

-- RAM Control State Machine
rdatreq_h.clk    = ckr_r;
rdatreq_h.d      = !qread_h & ramack_h;

rcSM.clk     = ckr_r;
rcSM.reset   = !clr_l;
CASE rcSM IS
    WHEN rc00 =>         -- <nothing> active
        IF rdatreq_h THEN rcSM = rc01;
        ELSE rcSM = rc02;
        END IF;
    WHEN rc01 =>         -- rdatoe_l active
```

```
        IF !rdatreq_h THEN rcSM = rc00;
        END IF;
    WHEN rc02 =>          -- roe_l active
        IF rdatreq_h THEN rcSM = rc00;
        END IF;
END CASE;

-- Receive Hotlink Input Registers
rdin_h[].clk     = ckr_r;
rdin_h[].d       = rd_h[];
rscin_h.clk      = ckr_r;
rscin_h.d        = rsc_h;
rvsin_h.clk      = ckr_r;
rvsin_h.d        = rvs_h;
rdyin_l.clk      = ckr_r;
!rdyin_l.d       = !rdy_l;

-- LED PWM's
color_h[].clk        = clk_r;
!color_h[].clrn      = !clr_l;
IF colorclr_h.q THEN
    IF !rgbcycle_h.q THEN
        color_h[0].d = (redreg_h[].q != 0);
        color_h[1].d = (grnreg_h[].q != 0);
        color_h[2].d = (blureg_h[].q != 0);
    ELSE
        color_h[].d = B"001";
    END IF;
ELSIF rgbcycle_h.q & nextcolor_h THEN
    color_h[1].d = color_h[0].q;
    color_h[2].d = color_h[1].q;
    color_h[0].d = color_h[2].q;
ELSE
    color_h[].d = color_h[].q;
END IF;

-- 'colorclr_h' is active if chSM is in state chIdle and it is going to stay in chIdle.
colorclr_h.clk       = clk_r;
!colorclr_h.clrn     = !clr_l;

enred_h      = color_h[0] & !blanksync_h;
engrn_h      = color_h[1] & !blanksync_h;
enblu_h      = color_h[2] & !blanksync_h;

redled.(clk_r, clr_l, enable_h, ratio_h[]) = (clk_r, clr_l, enred_h, redreg_h[].q);
redon_h = redled.on_h;

grnled.(clk_r, clr_l, enable_h, ratio_h[]) = (clk_r, clr_l, engrn_h, grnreg_h[].q);
grnon_h = grnled.on_h;

bluled.(clk_r, clr_l, enable_h, ratio_h[]) = (clk_r, clr_l, enblu_h, blureg_h[].q);
bluon_h = bluled.on_h;

jmpclr_l.clk     = clk_r;
!jmpclr_l.clrn   = !clr_l;
jmpclr_l.d       = VCC;
bistout_h        = bistin_h & jmpclr_l;
bypassout_h      = bypassin_h & jmpclr_l;
!rselb_l         = bypassin_h;

-- Diagnostic LED's
-- Err LED (red) is on if seqerr_h or aderrin_h are active or if errcnt_h[] is non-zero
-- Spare LED (yellow) is controlled by 'spare_h'
-- RBsy LED (green) is a pulse stretched version of 'rdy_l'. Goal is to be able to
```

```
    --   see LED change intensity if 'rdy_l' is on as little as 1/256 of the time,
    --   or, conversely, see an intensity drop if 'rdy_l' goes from being on all
    --   the time to 255/256 of the time.
    -- TBsy LED (green) is a pulse stretched version of "!tena_l # !tenn_l".
    !error_l     = seqerr_h # aderrin_h # (errcnt_h[] != 0);
    oneshot1.(clk_r, clr_l, in_l) = (ckr_r, clr_l, rdyin_l);
    !rbusy_l     = !oneshot1.out_l;
    oneshot2.(clk_r, clr_l, in_l) = (clk_r, clr_l, tena_l & tenn_l);
    !tbusy_l     = !oneshot2.out_l;
--  !tbusy_l     = !tena_l # !tenn_l;

    -- Delay Value (used to reduce fan-in requirements for 'delay_h[]')
--  delayvalue_h[].clk       = clk_r;
--  !delayvalue_h[].clrn     = !clr_l;
    IF rgbcycle_h THEN
        CASE color_h[] IS
            WHEN B"001" => delayvalue_h[] = reddelay_h[].q;
            WHEN B"010" => delayvalue_h[] = grndelay_h[].q;
            WHEN B"100" => delayvalue_h[] = bludelay_h[].q;
        END CASE;
    ELSE
        delayvalue_h[] = reddelay_h[].q;
    END IF;

    -- Delay Counters
    delay_h[].clk        = clk_r;
    !delay_h[].clrn      = !clr_l;
    IF delayld_h THEN
        delay_h[].d = delayvalue_h[];
    ELSIF delaycnt_h THEN
        delay_h[].d = delay_h[].q - 1;
    ELSE
        delay_h[].d = delay_h[].q;
    END IF;
    delaytc_h = (delay_h[] == 0);        -- implemented as an LCELL

    -- Millisecond delay counter (part 1)
    -- one millisecond = (33,332 + 1) x 30.0003 ns (33.3330 MHz) = H"8234"
    -- ten microseconds = (332.33 + 1) x 30.0003 ns (33.3330 MHz) = H"014C"
    msdelay_h[].clk      = clk_r;
    !msdelay_h[].clrn    = !clr_l;
    msdelayld_h          = delaycnt_h;
    IF msdelayld_h THEN
        IF linesync_h THEN
--          msdelay_h[07..00].d = 3;        -- simulating
            msdelay_h[07..00].d = H"4C";    -- LS byte of H"014C" or 332
        ELSE
--          msdelay_h[07..00].d = 7;        -- simulating
            msdelay_h[07..00].d = H"34";    -- LS byte of H"8234" or 33332
        END IF;
    ELSE
        msdelay_h[07..00].d = msdelay_h[07..00].q - 1;
    END IF;
    msdelaytc_h.clk      = clk_r;
    !msdelaytc_h.clrn    = !clr_l;
    msdelaytc_h.d        = (msdelaytc1_h & (msdelay_h[07..00] == 1));
    borrow_h             = (msdelay_h[07..00] == 0);     -- implemented as an LCELL

    -- Millisecond delay counter (part 2)
    IF msdelayld_h THEN
        IF linesync_h THEN
--          msdelay_h[15..08].d = 0;        -- simulating
            msdelay_h[15..08].d = H"01";    -- MS byte of H"014C"
        ELSE
```

```
--          msdelay_h[15..08].d = 0;          -- simulating
            msdelay_h[15..08].d = H"82";      -- MS byte of H"8234" or 33332
        END IF;
    ELSIF borrow_h THEN
        msdelay_h[15..08].d = msdelay_h[15..08].q - 1;
    ELSE
        msdelay_h[15..08].d = msdelay_h[15..08].q;
    END IF;
    msdelaytc1_h = (msdelay_h[15..08] == 0);    -- implemented as an LCELL

    -- Delay State Machine
    dlSM.clk    = clk_r;
    dlSM.reset  = !clr_l;

    CASE dlSM IS
        WHEN dl00 =>          -- delayld_h active
            IF dodelay_h THEN
                IF !repeat_h # delaytc_h THEN dlSM = dl03;
                ELSE dlSM = dl01;
                END IF;
            END IF;
        WHEN dl01 =>          -- delaycnt_h (msdelayld_h) active
            dlSM = dl02;
        WHEN dl02 =>          -- <nothing> active (msdelay_h[] counting)
            IF msdelaytc_h THEN
                IF delaytc_h THEN dlSM = dl03;
                ELSE dlSM = dl01;
                END IF;
            END IF;
        WHEN dl03 =>          -- delaydone_h active
            IF !dodelay_h THEN dlSM = dl00;
            END IF;
    END CASE;

    -- Receive Hotlink Data Descriptor Byte
    rdesc_h[].clk   = ckr_r;
    !rdesc_h[].clrn = !clr_l;
    IF rdescen_h & !rdyin_l & !rscin_h THEN
        rdesc_h[].d = rdin_h[];
    ELSE
        rdesc_h[].d = rdesc_h[].q;
    END IF;

    CASE rdesc_h[] IS
        WHEN CtrlStat =>
            ratc_h[] = 0;
            ctrlstatwe_h = we_h;
        WHEN ErrCnt =>
            ratc_h[] = 0;
            errcntwe_h = we_h;
        WHEN Delay =>
            ratc_h[] = 2;
            CASE rra_h[01..00] IS
                WHEN B"00" => reddelaywe_h = we_h;
                WHEN B"01" => grndelaywe_h = we_h;
                WHEN B"10" => bludelaywe_h = we_h;
            END CASE;
        WHEN LED =>
            ratc_h[] = 2;
            CASE rra_h[01..00] IS
                WHEN B"00" => redregwe_h = we_h;
                WHEN B"01" => grnregwe_h = we_h;
                WHEN B"10" => bluregwe_h = we_h;
            END CASE;
```

```
        WHEN LUT =>
--          ratc_h[] = 7;        -- simulating
            ratc_h[] = 8191;
        WHEN OTHERS =>
            ratc_h[] = 0;
    END CASE;

%=========================================================================
Read Hotlink State Machine
=========================================================================%
rhSM.clk     = ckr_r;
rhSM.reset   = !clr_l;

rhbistsync_h.clk      = ckr_r;
!rhbistsync_h.clrn    = !clr_l;
rhbistsync_h.d        = bistin_h;

rdacksync_h.clk       = ckr_r;
!rdacksync_h.clrn     = !clr_l;
rdacksync_h.d         = rdack_h;

!rbisten_l            = LCELL (bisten_h);      -- avoid NOT gate pushback

syncbyte_h            = rscin_h & (rdin_h[] == Sync);

CASE rhSM IS
    -- Wait for a command symbol
    WHEN rhIdle =>                   -- <nothing> active
        IF rhbistsync_h THEN rhSM = rhBIST1;
        ELSIF !rdyin_l THEN
            IF !rscin_h THEN setseqerr_h = VCC;
            ELSIF LCELL (rdin_h[] == WriteHdr) THEN rhSM = rhWrite;
            ELSIF LCELL (rdin_h[] == ReadHdr) THEN rhSM = rhRead;
            END IF;
        END IF;

    -- Wait for a data selector byte
    WHEN rhWrite =>                  -- rdescen_h active
        IF !rdyin_l THEN
            IF rvsin_h THEN rhSM = rhIdle;
            ELSIF !syncbyte_h THEN
                IF rscin_h THEN rhSM = rhIdle; setseqerr_h = VCC;
                ELSIF LCELL (rdin_h[] == LUT) THEN rhSM = rhLUTData;
                ELSE rhSM = rhRegData;
                END IF;
            END IF;
        END IF;

    -- Write a byte to the LUT...
    WHEN rhLUTData =>                -- <nothing> active
        IF !rdyin_l THEN
            rdatoutld_h = VCC;
            IF rvsin_h THEN rhSM = rhIdle;
            ELSIF !syncbyte_h THEN
                IF rscin_h THEN rhSM = rhIdle; setseqerr_h = VCC;
                ELSE rhSM = rhWrLUT;
                END IF;
            END IF;
        END IF;
    WHEN rhWrLUT =>                  -- <nothing> active
        rwe_h = VCC;
        rrainc_h = VCC;
        IF rratc_h THEN rhSM = rhWrEndPkt;
        ELSE rhSM = rhLUTData;
```

```
    END IF;

-- ... or, write a byte to an internal register
WHEN rhRegData =>           -- <nothing> active
    IF !rdyin_l THEN
        IF rvsin_h THEN rhSM = rhIdle;
        ELSIF !syncbyte_h THEN
            IF rscin_h THEN rhSM = rhIdle; setseqerr_h = VCC;
            ELSE
                we_h = VCC;
                rrainc_h = VCC;
                IF rratc_h THEN rhSM = rhWrEndPkt;
                END IF;
            END IF;
        END IF;
    END IF;

-- Wait for EndPkt symbol
WHEN rhWrEndPkt =>          -- <nothing> active
    IF !rdyin_l THEN
        IF rvsin_h THEN rhSM = rhIdle;
        ELSIF !syncbyte_h THEN
            IF !rscin_h # (rdin_h[] != EndPkt) THEN setseqerr_h = VCC;
            END IF;
            rhSM = rhIdle;
        END IF;
    END IF;

--------------------------------------------------------------------------------
-- Wait for data selector byte
WHEN rhRead =>              -- rdescen_h active
    IF !rdyin_l THEN
        IF rvsin_h THEN rhSM = rhIdle;
        ELSIF !syncbyte_h THEN
            IF rscin_h THEN rhSM = rhIdle; setseqerr_h = VCC;
            ELSE rhSM = rhRdEndPkt;
            END IF;
        END IF;
    END IF;

-- Wait for EndPkt symbol
WHEN rhRdEndPkt =>          -- <nothing> active
    IF !rdyin_l THEN
        IF rvsin_h THEN rhSM = rhIdle;
        ELSIF !syncbyte_h THEN
            IF !rscin_h # (rdin_h[] != EndPkt) THEN rhSM = rhIdle; setseqerr_h = VCC;
            ELSE rhSM = rhQRead;
            END IF;
        END IF;
    END IF;

-- Queue the read
WHEN rhQRead =>             -- qread_h active
    IF !rdyin_l THEN
        IF rvsin_h THEN rhSM = rhIdle;
        ELSE rhSM = rhIdle; setseqerr_h = VCC;
        END IF;
    ELSIF rdacksync_h THEN
        rhSM = rhIdle;
    END IF;

--------------------------------------------------------------------------------
-- BIST mode entry, disable error reporting, load millisecond counter
WHEN rhBIST1 =>             -- noerrcnt_h (msdelayld_h) active
```

```
        msdelayld_h = VCC;
        rhSM = rhBIST2;

    -- Wait a millisecond for the loop to stabilize
    WHEN rhBIST2 =>             -- noerrcnt_h, bisten_h active
        IF msdelaytc_h THEN rhSM = rhBIST3;
        END IF;

    -- Enable error counter and wait for BIST to end
    WHEN rhBIST3 =>             -- bisten_h active
        IF !rhbistsync_h THEN rhSM = rhBIST4;
        END IF;

    -- disable error reporting, load millisecond counter
    WHEN rhBIST4 =>             -- noerrcnt_h, bisten_h (msdelayld_h) active
        msdelayld_h = VCC;
        rhSM = rhBIST5;

    -- Turn off BIST and wait a millisecond for the loop to stabilize
    WHEN rhBIST5 =>             -- noerrcnt_h active
        IF msdelaytc_h THEN rhSM = rhIdle;
        END IF;

END CASE;

%=====================================================================
Transmit Hotlink Data Multiplexor
=====================================================================%

-- Transmit Hotlink Data Descriptor Byte
tdesc_h[].clk   = clk_r;
!tdesc_h[].clrn = !clr_l;
IF video_h THEN
    tdesc_h[7].d = GND;
    tdesc_h[6..4].d = color_h[2..0].q;
    tdesc_h[3].d = GND;
    IF tceven_h THEN tdesc_h[2..0].d = Even;
    ELSIF tcgo_h THEN tdesc_h[2..0].d = Odd;
    ELSE tdesc_h[2..0].d = Line;
    END IF;
ELSE
    tdesc_h[7..3].d = GND;
    tdesc_h[2..0].d = rdesc_h[2..0].q;
END IF;

-- We need 'regout_h[]' to reduce the fan-in requirements of 'tdout_h[]'.
CASE tdesc_h[] IS
    WHEN CtrlStat =>
        regout_h[7] = !spare_l;
        regout_h[6] = rf_h.q;
        regout_h[5] = allpixels_h;
        regout_h[4] = rgbcycle_h;
        regout_h[3] = repeat_h;
        regout_h[2] = line_h;
        regout_h[1] = odd_h;
        regout_h[0] = even_h;
    WHEN ErrCnt =>
        regout_h[7] = seqerr_h;
        regout_h[6] = aderrin_h;
        regout_h[5..0] = errcnt_h[];
END CASE;

CASE seq_h[] IS
    WHEN tDataHdr =>
```

```
                tdout_h[].d = DataHdr;
                tscout_h.d = VCC;
            WHEN tDataDesc =>
                tdout_h[].d = tdesc_h[];
                tscout_h.d = GND;
            WHEN tData =>
                CASE tdesc_h[] IS
                    WHEN Delay =>
                        CASE tra_h[01..00] IS
                            WHEN B"00" => tdout_h[].d = reddelay_h[];
                            WHEN B"01" => tdout_h[].d = grndelay_h[];
                            WHEN B"10" => tdout_h[].d = bludelay_h[];
                        END CASE;
                    WHEN LED =>
                        CASE tra_h[01..00] IS
                            WHEN B"00" => tdout_h[].d = redreg_h[];
                            WHEN B"01" => tdout_h[].d = grnreg_h[];
                            WHEN B"10" => tdout_h[].d = blureg_h[];
                        END CASE;
                    WHEN LUT =>
                        tdout_h[].d = rdat_h[];
                    WHEN OTHERS =>
                        tdout_h[].d = regout_h[];
                END CASE;
                tscout_h.d = GND;
            WHEN tEndPkt =>
                tdout_h[].d = EndPkt;
                tscout_h.d = VCC;
        END CASE;

        -- Transmit Hotlink Output Signals
        tscout_h.clk    = clk_r;
        !tscout_h.clrn  = !clr_l;
        tsc_h           = tscout_h.q;

        tdout_h[].clk   = clk_r;
        !tdout_h[].clrn = !clr_l;
    —   tdout_h[].ena   = !tenn_l;
        tdtri_h[].in    = tdout_h[].q;
        tdtri_h[].oe    = !tdtrioe_l;
        td_h[]          = tdtri_h[].out;

        %==================================================================
        Transmit Hotlink State Machine
        ==================================================================%
        -- We allow for an abort by writing to the control/stat register with
        -- bits even_h, odd_h, and line_h zero. This is a special case in order
        -- to avoid having to cycle the power to the camera head when things go
        -- VERY wrong. This clears the thSM and chSM. If video was being
        -- transmitted at the time (eg. 'repeat_h' active), you'll probably get
        -- a Tx FIFO overflow. The normal way to stop live video is to remove
        -- just the 'repeat_h' bit and let the other bits clear themselves.
        ctrlstatwed_h.clk   = ckr_r;
        !ctrlstatwed_h.clrn = !clr_l;
        ctrlstatwed_h.d     = ctrlstatwe_h;

        !clrsm_l    = !gclr_l # (ctrlstatwed_h & !even_h & !odd_h & !line_h);    -- LCELL

        thSM.clk    = clk_r;
        thSM.reset  = !clrsm_l;

        qreadsync_h.clk     = clk_r;
        !qreadsync_h.clrn   = !clr_l;
        qreadsync_h.d       = qread_h;
```

```
-- 'video_h' is a latched version of 'senddesc_h' used to record the type of
-- packet we're sending. Note that register reads (aka 'qreadsync_h')
-- have priority over video. Once a packet transmission starts, 'video_h'
-- keeps us from switching over to another packet type DURING transmission.
video_h.clk         = clk_r;
!video_h.clrn       = !clr_l;
video_h.ena         = (thSM == thIdle);
video_h.d           = senddesc_h & !qreadsync_h;

qbistsync_h.clk     = clk_r;
!qbistsync_h.clrn   = !clr_l;
qbistsync_h.d       = bisten_h;

tbisten_l.clk       = clk_r;
!tbisten_l.prn      = !clr_l;

lutsel_h    = (tdesc_h[] == LUT);    -- implemented as an LCELL

CASE thSM IS
    WHEN thIdle =>          -- seq_h[] = tDataDesc, tdtrioe_l active
        IF qreadsync_h # senddesc_h THEN thSM = thDatHdr;
        ELSIF qbistsync_h THEN thSM = thBIST; tbisten_l.r = VCC;
        END IF;

    -- Send DataHdr symbol
    WHEN thDatHdr =>        -- seq_h[] = tDataHdr, traclr_h, tenn_l, tdtrioe_l active
        thSM = thDatDesc;

    -- Send data descriptor byte
    WHEN thDatDesc =>       -- seq_h[] = tDataDesc, tenn_l, tdtrioe_l active
        IF video_h THEN thSM = thDoVid;
        ELSIF lutsel_h THEN thSM = thRdLUT;
        ELSE thSM = thDoRd;
        END IF;

    -- Address the LUT RAM
    WHEN thRdLUT =>         -- seq_h[] = tData, tdtrioe_l active
        thSM = thDoRd;

    -- Send a data byte
    WHEN thDoRd =>          -- seq_h[] = tData, trainc_h, tenn_l, tdtrioe_l active
        IF tratc_h THEN thSM = thEndRd;
        ELSIF lutsel_h THEN thSM = thRdLUT;
        END IF;
    WHEN thEndRd =>         -- seq_h[] = tEndPkt, rdack_h, tdtrioe_l active
        IF !qreadsync_h THEN thSM = thSndEndPkt;
        END IF;

    -- Send EndPkt symbol
    WHEN thSndEndPkt =>     -- seq_h[] = tEndPkt, tenn_l, tdtrioe_l active
        thSM = thIdle;

    ------------------------------------------------------------------------------
    -- Send video data (allow tdtrioe_l to be active for a cycle to send data)
    WHEN thDoVid =>         -- seq_h[] = tDataDesc, tdtrioe_l active
        thSM = thVidDat1;
    -- Hi-Z 'td_h[]' outputs
    WHEN thVidDat1 =>       -- seq_h[] = tDataDesc active
        thSM = thVidDat2;
    -- Lo-Z transmit FIFO outputs and enable reading when FIFO is non-empty
    WHEN thVidDat2 =>       -- seq_h[] = tDataDesc, entfren_l active
        IF sendtrailer_h THEN thSM = thVidDat3;
        END IF;
```

```
    -- Hi-Z transmit FIFO outputs, disable reading
    WHEN thVidDat3 =>         -- seq_h[] = tDataDesc active
        thSM = thSndEndPkt;

    -----------------------------------------------------------------------------
    -- Activate 'tenn_l' for BIST mode
    WHEN thBIST =>            -- seq_h[] = tDataDesc, tenn_l, tdtrioe_l (tbisten_l) active
        IF !qbistsync_h THEN thSM = thIdle; tbisten_l.s = VCC;
        END IF;
END CASE;

%==========================================================================
Camera Head State Machine
==========================================================================%
chSM.clk     = clk_r;
chSM.reset   = !clrsm_l;

evensync_h.clk       = clk_r;
!evensync_h.clrn     = !clr_l;
evensync_h.d         = even_h.q;

oddsync_h.clk        = clk_r;
!oddsync_h.clrn      = !clr_l;
oddsync_h.d          = odd_h.q;

linesync_h.clk       = clk_r;
!linesync_h.clrn     = !clr_l;
linesync_h.d         = line_h.q;

busysync_h.clk       = clk_r;
!busysync_h.clrn     = !clr_l;
busysync_h.d         = busy_h;

blanksync_h.clk      = clk_r;
!blanksync_h.clrn    = !clr_l;
blanksync_h.d        = blank_h;

CASE chSM IS
    WHEN chIdle =>            -- (colorclr_h) may be active
        IF evensync_h THEN chSM = chEvenDelay;
        ELSIF oddsync_h THEN chSM = chOddDelay;
        ELSIF linesync_h THEN chSM = chLineDelay;
        ELSE colorclr_h.d = VCC;
        END IF;

    -- Send Even Field.
    WHEN chEvenDelay =>       -- dodelay_h active
        IF delaydone_h THEN chSM = chEven1;
        END IF;
    WHEN chEven1 =>           -- tceven_h, tcgo_h active
        IF blanksync_h THEN chSM = chEven2;
        END IF;
    WHEN chEven2 =>           -- tceven_h, tcgo_h, senddesc_h active
        IF !entfren_l THEN chSM = chEven3;
        END IF;
        clreven_h = !repeat_h.q;
    WHEN chEven3 =>           -- <nothing> active
        IF !busysync_h & tfor_l THEN
            IF oddsync_h THEN chSM = chTrailer;
            ELSE chSM = chColor1;
            END IF;
        END IF;
    WHEN chTrailer =>         -- sendtrailer_h active
        IF thSM == thIdle THEN chSM = chOddDelay;
```

```
        END IF;

    -- Send Odd Field.
    WHEN chOddDelay =>          -- dodelay_h active
        IF delaydone_h THEN chSM = chOdd1;
        END IF;
    WHEN chOdd1 =>              -- tcgo_h active
        IF blanksync_h THEN chSM = chOdd2;
        END IF;
    WHEN chOdd2 =>              -- tcgo_h, senddesc_h active
        IF !entfren_l THEN chSM = chOdd3;
        END IF;
        clrodd_h = !repeat_h.q;
    WHEN chOdd3 =>              -- <nothing> active
        IF !busysync_h & tfor_l THEN chSM = chColor1;
        END IF;

    -- Send Line Field.
    WHEN chLineDelay =>         -- dodelay_h active
        IF delaydone_h THEN chSM = chLine1;
        END IF;
    WHEN chLine1 =>             -- rlgo_h active
        IF blanksync_h THEN chSM = chLine2;
        END IF;
    WHEN chLine2 =>             -- rlgo_h, senddesc_h active
        IF !entfren_l THEN chSM = chLine3;
        END IF;
        clrline_h = !repeat_h.q;
    WHEN chLine3 =>             -- <nothing> active
        IF !busysync_h & tfor_l THEN chSM = chColor1;
        END IF;

    -- Send trailer and proceed to next color
    WHEN chColor1 =>            -- sendtrailer_h, nextcolor_h active
        chSM = chColor2;
    WHEN chColor2 =>            -- sendtrailer_h active
        IF thSM == thIdle THEN chSM = chIdle;
        END IF;
END CASE;

%=========================================================================
Transmit FIFO Configuration State Machine
=========================================================================%

tcclkselin_h.clk    = clk_r;
!tcclkselin_h.clrn  = !clr_l;
tcclkselin_h.d      = tcclksel_h;

tfSM.clk    = clk_r;
tfSM.reset  = !clrsm_l;

CASE tfSM IS
    -- Reset FIFO pointers on powerup or abort
    WHEN tfAlmostRL =>          -- tfprs_l active
        IF tcclkselin_h THEN tfSM = tfAlmostTC;
        ELSE tfSM = tfRL;
        END IF;

    -- RL Clock is selected
    WHEN tfRL =>                -- <nothing> active
        IF tcclkselin_h THEN tfSM = tfAlmostRL;
        END IF;

    WHEN tfAlmostTC =>          -- tfprs_l, tffs_h active
        IF tcclkselin_h THEN tfSM = tfTC;
```

```
            ELSE tfSM = tfAlmostRL;
            END IF;

        -- TC Clock is selected
        WHEN tfTC =>            -- tffs_h active
            IF !tcclkselin_h THEN tfSM = tfAlmostTC;
            END IF;
    END CASE;
END;
```

```
%===========================================================================
Pulse Stretcher

Kensal Corporation
OSC Project

Author: Ken Crocker
Date:   29 Sep 96
File:   STRETCH.TDF
Rev:    1.0

The goal of this module is to pulse stretch a signal so that an insignificant
duty cycle is increased to become significant. For example, say one wants
to observe the 'rdy_l' signal and it only goes active 1/256 of the time.
This routine will boost the duty cycle to 32/256, which will be observable.
Conversely, say the 'rdy_l' signal is on 255/256 of the time. This routine
will decrease the duty cycle to 224/256, which should be observable against
an LED that is on all the time (ie 256/256).
===========================================================================%
TITLE "Pulse Stretcher";

SUBDESIGN stretch (
    clk_r                   : INPUT;
    clr_l                   : INPUT;

    in_l                    : INPUT;
    out_l                   : OUTPUT;
)

VARIABLE
    out_l                   : SRFF;
    ind1_l                  : DFF;
    ind2_l                  : DFF;
    cntclr_h                : LCELL;
    cnt_h[4..0]             : DFF;
    cnttc_h                 : LCELL;

BEGIN
    -- 'in_l' register
    ind1_l.clk      = clk_r;
    !ind1_l.prn     = !clr_l;
    !ind1_l.d       = !in_l;

    -- delay register
    ind2_l.clk      = clk_r;
    !ind2_l.prn     = !clr_l;
    !ind2_l.d       = !ind1_l;

    out_l.clk       = clk_r;
    !out_l.prn      = !clr_l;
    out_l.s         = cnttc_h &  ind1_l;
    out_l.r         = cnttc_h & !ind1_l;

    -- 'in_l' edge detector
    cntclr_h        = (cnt_h[] == H"1F") & ((!ind1_l & ind2_l) # (ind1_l & !ind2_l));   -- LCELL

    -- Pulse stretcher
    cnt_h[].clk     = clk_r;
    !cnt_h[].clrn       = !clr_l;
    IF cntclr_h THEN
        cnt_h[].d = 0;
    ELSIF !cnttc_h THEN
        cnt_h[].d = cnt_h[].q + 1;
    ELSE
```

```
        cnt_h[].d = cnt_h[].q;
    END IF;
    cnttc_h     = (cnt_h[] == H"1F");        -- LCELL
END;
```

```
%=============================================================================
Anti-blooming Gate Controller

Kensal Corporation
RL4096P Linescan Array Controller

Upon reciept of a 'go_h' signal, this module will activate 'ab_h' for 750 us,
simulating movement of the slide to the next location, followed by an
inactive 'ab_h' for 124 us, which simulates the time required for the first
exposure.

Author: Ken Crocker
Date:   12 Feb 96
File:   RLABGC.TDF
Rev:    1.0
=============================================================================%
TITLE "RL4096P Linescan Array Controller - Anti-Blooming Gate Controller";

SUBDESIGN rlabgc
(
    gclk_r          : INPUT;    -- 33.3333 MHz clock (30ns period)
    gclr_l          : INPUT;
    simulating_h    : INPUT;

    go_h            : INPUT;

    ab_h            : OUTPUT;

    busy_h          : OUTPUT;
)

VARIABLE
    clk_r           : NODE;
    clr_l           : NODE;

    cnt[14..00]     : DFF;
    cnttc1_h        : NODE;
    cnttc2_h        : NODE;

    abgsm           : MACHINE OF BITS (ab_h, cnten_h, busy_h)
                        WITH STATES (
                            a00 = B"000",
                            a01 = B"111",
                            a02 = B"001",
                            a03 = B"011"
                        );

BEGIN
    clk_r       = GLOBAL (gclk_r);
    clr_l       = GLOBAL (gclr_l);

    cnt[].clk   = clk_r;
    cnt[].clrn  = clr_l;
    IF (cnten_h) THEN
        cnt[].d = cnt[].q + 1;
    ELSE
        cnt[].d = 0;
    END IF;

    IF !simulating_l THEN
        cnttc1_h = (cnt[] == 250);      -- shorten to 7.5 us to make simulation easier
        cnttc2_h = (cnt[] == 41);       -- shorten to 1.24 us to make simulation easier
    ELSE
        cnttc1_h = (cnt[] == 24999);    -- normal delay is 750 us.
```

```
        cnttc2_h = (cnt[] == 4132);       -- normal delay is 124 us.
    END IF;

    abgsm.clk   = clk_r;
    abgsm.reset = !clr_l;

%=========================================================================
    Anti-Blooming Gate State Machine
=========================================================================%
    CASE (abgsm) IS
        WHEN a00 =>
            IF go_h THEN abgsm = a01;
            END IF;

        WHEN a01 =>                      -- ab_h active
            IF cnttc1_h THEN abgsm = a02;
            END IF;

        WHEN a02 =>
            abgsm = a03;

        WHEN a03 =>
            IF cnttc2_h THEN abgsm = a00;
            END IF;
    END CASE;

END;
```

```
%===========================================================================
CCD State Machine Controller

Kensal Corporation
RL4096P Linescan Array Controller

Author: Ken Crocker
Date:   15 Feb 96
File:   RLCCDSMC.TDF
Rev:    1.00
===========================================================================%
TITLE "RL4096P Linescan Array Controller - CCD State Machine Controller";

SUBDESIGN rlccdsmc
(
    gclk_r          : INPUT;
    gclr_l          : INPUT;

    go_h            : INPUT;    -- may be asynchronous w.r.t. 'gclk_r'
    sync_h          : INPUT;

    hztmgbusy_h     : INPUT;
    readoutbusy_h   : INPUT;

    newcolor_h      : OUTPUT;
    readline_h      : OUTPUT;

    busy_h          : OUTPUT;
    hsync_h         : OUTPUT;
)

VARIABLE
    clk_r           : NODE;
    clr_l           : NODE;

    gosync_h        : DFF;      -- synchronizing F/F
    ccdsm           : MACHINE OF BITS (newcolor_h, readline_h, hsync_h, busy_h)
                        WITH STATES (
                            c00 = B"0000",
                            c01 = B"1011",
                            c02 = B"0011",
                            c03 = B"0101",
                            c04 = B"0001"
                    );

BEGIN
    clk_r    = GLOBAL (gclk_r);
    clr_l    = GLOBAL (gclr_l);

    gosync_h.clk    = clk_r;
    !gosync_h.clrn  = !clr_l;
    gosync_h.d      = go_h;

%===========================================================================
    CCD State Machine
===========================================================================%
    ccdsm.clk    = clk_r;
    ccdsm.reset = !clr_l;

    CASE ccdsm IS
        WHEN c00 =>             -- <nothing> active
            IF gosync_h.q & sync_h THEN ccdsm = c01;
            END IF;
        WHEN c01 =>             -- newcolor_h, busy_h active
```

```
            IF hztmgbusy_h THEN ccdsm = c02;
            END IF;
        WHEN c02 =>                -- busy_h active
            IF !hztmgbusy_h THEN ccdsm = c03;
            END IF;
        WHEN c03 =>                -- readline_h, busy_h active
            IF readoutbusy_h THEN ccdsm = c04;
            END IF;
        WHEN c04 =>                -- busy_h active
            IF !readoutbusy_h THEN ccdsm = c00;
            END IF;
    END CASE;

END;
```

```
%=====================================================================
1/2 Clock Delay Module

Kensal Corporation
RL4000P Linescan Array Controller

Author: Ken Crocker
File:   RLDELAY.TDF

Revision History:
    1.00    20 Nov 96   K. W. Crocker
        Initial writing.
=====================================================================%
TITLE "RL4096P Linescan Array Controller - 1/2 Clock Delay Module";

SUBDESIGN rldelay
(
    gclk_r      : INPUT;
    in_h        : INPUT;
    out_h       : OUTPUT;
)

VARIABLE
    clk_r       : NODE;
    in2_h       : DFF;

BEGIN
    clk_r       = GLOBAL (gclk_r);

    in2_h.clk   = !clk_r;
    in2_h.d     = in_h;
    out_h       = in2_h;
END;
```

```
%===========================================================================
Horizontal Timing Controller

Kensal Corporation
RL4096P Linescan Array Controller

Author: Ken Crocker
Date:   12 Feb 95
File:   HZTMGC.TDF
Rev:    1.0
===========================================================================%
TITLE "RL4096P Linescan Array - Horizontal Timing Controller";

SUBDESIGN rlhztmgc
(
    gclk_r      : INPUT;    -- 33.3 MHz clock input (30ns period)
    gclr_l      : INPUT;

    go_h        : INPUT;

    tg_h        : OUTPUT;
    pg_l        : OUTPUT;

    busy_h      : OUTPUT;
)

VARIABLE
    clk_r       : NODE;
    clr_l       : NODE;

    cnt[2..0]   : DFF;
    cnttc_h     : NODE;

-- State Definitions
    htsm        : MACHINE OF BITS (
                      tg_h, pg_l, cnten_h, busy_h
                  ) WITH STATES (
                          h00 = B"0100",
                          h01 = B"0101",
                          h02 = B"1111",
                          h03 = B"1011",
                          h04 = B"0011"
                  );

BEGIN
    clk_r       = GLOBAL (gclk_r);
    clr_l       = GLOBAL (gclr_l);

    cnt[].clk   = clk_r;
    cnt[].clrn  = clr_l;
    IF cnten_h & !cnttc_h THEN
        cnt[].d = cnt[].q + 1;
    ELSE
        cnt[].d = 0;
    END IF;
    cnttc_h = cnt[] == B"100";

    htsm.clk    = clk_r;
    htsm.reset  = !clr_l;

%===========================================================================
    Horizontal Timing State Machine
===========================================================================%
    CASE (htsm) IS
```

```
        WHEN h00 =>             -- <nothing> active
            IF go_h THEN
                htsm = h01;
            END IF;

        WHEN h01 =>             -- <nothing> active
            htsm = h02;

        WHEN h02 =>             -- tg_h, busy_h active
            IF cnttc_h THEN
                htsm = h03;
            END IF;

        WHEN h03 =>             -- tg_h, pg_l, busy_h active
            IF cnttc_h THEN
                htsm = h04;
            END IF;

        WHEN h04 =>             -- pg_l, busy_h active
            IF cnttc_h THEN
                htsm = h00;
            END IF;
    END CASE;
END;
```

```
%==============================================================
CCD Controller, Top Level

Kensal Corporation
EG&G RL4096P Linescan Array Controller
RL4096P Evaluation Board

Author: Ken Crocker
Date:   14 Nov 96
File:   MAIN.TDF
Rev:    1.0
==============================================================%
TITLE "EG&G RL4096P Linescan Array Controller, Top Level";

INCLUDE "RLCCDSMC.INC";
INCLUDE "RLHZTMGC.INC";
INCLUDE "RLREADC.INC";

SUBDESIGN rlmain
(
    gclk_r              : INPUT;        -- 33.3333 MHz clock
    gclr_l              : INPUT;        -- active low asynchronous system reset
    oe_l                : INPUT;        -- reserved OE pin
    gclk2_r             : INPUT;        -- reserved GCLK pin

    -- Interface with RL4000P
    ab_h                : OUTPUT;       -- anti-blooming/exposure control
    tg_h                : OUTPUT;       -- transfer gate
    pg_l                : OUTPUT;       -- pixel gate
    rg_h                : OUTPUT;       -- reset gate
    srg_l               : OUTPUT;       -- CCD A & B, phase #1
    srg_h               : OUTPUT;       -- CCD A & B, phase #2

    -- Interface with Clamp
    clmp_h2             : OUTPUT;
    clmp_h1             : OUTPUT;

    -- Interface with A/D Controller
    rlgo_h              : INPUT;
    allpixels_h         : INPUT;
    rlclk_r             : OUTPUT;       -- 8.333 MHz clock to A/D Controller
    rlbusy_h            : OUTPUT;       -- LCELL delayed outputs to A/D Controller
    rlblank_h           : OUTPUT;       -- LCELL delayed outputs to A/D Controller
    rldv_h              : OUTPUT;       -- LCELL delayed outputs to A/D Controller

    -- Diagnostic signals
    hsync_h             : OUTPUT;
    simulating_l        : INPUT;
)

VARIABLE
    clk_r               : NODE;
    clr_l               : NODE;

    hztmgbusy_h         : NODE;
    readoutbusy_h       : NODE;

    newcolor_h          : NODE;
    readline_h          : NODE;

    hztmg               : rlhztmgc;
    readout             : rlreadc;
    ccdsm               : rlccdsmc;
```

```
    ckSM                : MACHINE OF BITS (
                              rlclk_r, sync_h
                          ) WITH STATES (
                              ck00 = B"00",
                              ck01 = B"00",
                              ck02 = B"10",
                              ck03 = B"10",
                              ck04 = B"00",
                              ck05 = B"00",
                              ck06 = B"11",
                              ck07 = B"10"
                          );

BEGIN
    clk_r           = GLOBAL (gclk_r);
    clr_l           = GLOBAL (gclr_l);

    -- Clock State Machine
    -- The signal 'rlclk_r' is actually twice the frequency of the two A/D clocks
    -- generated by the A/D Controller. In order to minimize the number of signal
    -- lines between the RL4000P and A/D Controllers we use 'rlbusy_h' to force
    -- synchronization of the A/D clocks. The A/D Controller then verifies sync
    -- when 'rldv_h' goes active. Therefore, there must be modulo 2 'rlclk_r'
    -- clocks between 'rlbusy_h' and 'rldv_h'. 'sync_h' will go active once every
    -- two 'rlclk_r' cycles, the 'clk_r' cycle immediately before 'rlclk_r' goes active.
    ckSM.clk        = clk_r;
    ckSM.reset      = !clr_l;

    CASE ckSM IS
        WHEN ck00 => ckSM = ck01;   -- <nothing> active
        WHEN ck01 => ckSM = ck02;   -- <nothing> active
        WHEN ck02 => ckSM = ck03;   -- rlclk_r active    (this is an A/D clock edge!)
        WHEN ck03 => ckSM = ck04;   -- rlclk_r active
        WHEN ck04 => ckSM = ck05;   -- <nothing> active
        WHEN ck05 => ckSM = ck06;   -- <nothing> active
        WHEN ck06 => ckSM = ck07;   -- rlclk_r, sync_h  (this is NOT an A/D clock edge)
        WHEN ck07 => ckSM = ck00;   -- rlclk_r active (busy_h/dv_h may go active on this cycle)
    END CASE;

    -- A/D Controller Interface (partial)
    -- See 'ckSM' above for 'rlclk_r' output.
    -- See Readout Controller for 'rldv_h' output.
    -- See CCD State Machine Controller for 'rlbusy_h' output.
    rlblank_h       = newcolor_h # hztmgbusy_h;

    -- Anti-blooming Control
    ab_h            = GND;

    -- Horizontal Timing
    hztmg.( gclk_r, gclr_l, go_h) = (gclk_r, gclr_l, newcolor_h);
    (tg_h, pg_l, hztmgbusy_h) = hztmg.( tg_h, pg_l, busy_h);

    -- Readout
    readout.( gclk_r, gclr_l, simulating_l, allpixels_h, go_h, sync_h)
        = (gclk_r, gclr_l, simulating_l, allpixels_h, readline_h, sync_h);
    (rg_h, srg_l, srg_h, clmp_h2, clmp_h1, rldv_h, readoutbusy_h)
        = readout.(rg_h, srg_l, srg_h, clmp_h2, clmp_h1, dv_h, busy_h);

    -- CCD State Machine
    ccdsm.( gclk_r, gclr_l, go_h, sync_h, hztmgbusy_h, readoutbusy_h)
        = (gclk_r, gclr_l, rlgo_h, sync_h, hztmgbusy_h, readoutbusy_h);
    (newcolor_h, readline_h, rlbusy_h, hsync_h)
        = ccdsm.(newcolor_h, readline_h, busy_h, hsync_h);
END;
```

```
%===========================================================================
Readout Controller

Kensal Corporation
RL4096P Linescan Array Controller

Author: Ken Crocker
Date:   13 Feb 96
File:   RLREADC.TDF

Revision History:
    1.00    12 Feb 96   K. W. Crocker
        Initial writing.
    1.01    13 Feb 96   K. W. Crocker
        Modify to use 33.3 MHz clock.
    1.02    7 Jun 96    K. W. Crocker
        Slow down to 4.166 MHz readout due to slow rise of Elantec drivers.
    1.03    20 Nov 96   K. W. Crocker
        Modify for A/D Controller
===========================================================================%
TITLE "RL4096P Linescan Array Controller - Readout Controller";

INCLUDE "rldelay.inc";

SUBDESIGN rlreadc
(
    gclk_r          : INPUT;    -- 33.3333 MHz clock input (30 ns period)
    gclr_l          : INPUT;
    simulating_l    : INPUT;

    allpixels_h     : INPUT;
    go_h            : INPUT;
    sync_h          : INPUT;

    rg_h            : OUTPUT;   -- reset gate
    srg_l           : OUTPUT;   -- phase 2 gate (aka shift register gate)
    srg_h           : OUTPUT;   -- phase 1 gate (aka shift register gate)

    clmp_h2         : OUTPUT;
    clmp_h1         : OUTPUT;

    dv_h            : OUTPUT;
    busy_h          : OUTPUT;
)

VARIABLE
    clk_r           : NODE;
    clr_l           : NODE;

    srgdelay1       : rldelay;
    srgdelay2       : rldelay;

    rosm            : MACHINE OF BITS (
                        rg_h, srgl_h, srgl_l, clmp_h2, clmp_h1, cnten_h, busy_h
                    ) WITH STATES(
                        r00 = B"0011100",
                        r01 = B"0011101",
                        r02 = B"0011101",
                        r03 = B"0011101",

                        r04 = B"0011101",
                        r05 = B"0010001",

                        r06 = B"0010001",
```

```
                        r07 = B"1010001",
                        r08 = B"1011101",
                        r09 = B"0101101",
                        r10 = B"0100001",
                        r11 = B"0100001",
                        r12 = B"0100011",

                        r13 = B"0011101"
                    );

    dv_h            : SRFF;

    cnt_h[11..00]   : DFF;
    cnttc_h         : NODE;

BEGIN
    clk_r   = GLOBAL (gclk_r);
    clr_l   = GLOBAL (gclr_l);

    cnt_h[].clk     = clk_r;
    !cnt_h[].clrn   = !clr_l;
    IF cnten_h THEN
        cnt_h[].d = cnt_h[].q + 1;
    ELSIF busy_h THEN
        cnt_h[].d = cnt_h[].q;
    ELSE
        cnt_h[].d = 0;
    END IF;

    dv_h.clk    = clk_r;
    !dv_h.clrn  = !clr_l;

    IF allpixels_h THEN
        dv_h.s = sync_h & (cnt_h[] == 3);       -- 3 dummy pixels
    ELSE
        dv_h.s = sync_h & (cnt_h[] == 19);      -- 3 dummies, 16 dark pixels
    END IF;
    IF !simulating_l THEN
        cnttc_h = LCELL (cnt_h[] == 34);        -- 0..34 = 35
                                                -- 35 - 19 = 16 "valid" pixel pairs (allpixels_h = GND)
                                                -- 35 -  3 = 32 "valid" pixel pairs (allpixels_h = VCC)
    ELSE
        cnttc_h = LCELL (cnt_h[] == 2066);      -- 0..2066 = 2067
                                                -- 2067 - 19 = 2048 "valid" pixel pairs (allpixels_h =
GND)
                                                -- 2067 -  3 = 2064 "valid" pixel pairs (allpixels_h =
VCC)
    END IF;

    -- Delay 'srg_h' by 1/2 clock
    srgdelay1.( gclk_r, in_h) = (gclk_r, srg1_h);
    (srg_h) = srgdelay1.( out_h);

    -- Delay 'srg_l' by 1/2 clock
    srgdelay2.( gclk_r, in_h) = (gclk_r, srg1_l);
    (srg_l) = srgdelay2.( out_h);

%=====================================================================
    Readout State Machine
=====================================================================%
    rosm.clk        = clk_r;
    rosm.reset      = !clr_l;

    CASE rosm IS
```

```
        WHEN r00 =>           -- <nothing> active
            IF go_h & sync_h THEN rosm = r01;
            END IF;
        WHEN r01 =>           -- clmp_h[], busy_h active (ckSM == ck07)
            rosm = r02;
        WHEN r02 =>           -- clmp_h[], busy_h active (ckSM == ck00)
            rosm = r03;
        WHEN r03 =>           -- clmp_h[], busy_h active (ckSM == ck01)
            rosm = r04;

        WHEN r04 =>           -- clmp_h[], busy_h active (ckSM == ck02, adclks)
            rosm = r06;
        WHEN r05 =>           -- busy_h active (ckSM == ck02, adclks)
            rosm = r06;

        WHEN r06 =>           -- clmp_h[], busy_h active (ckSM == ck03, adclks)
            rosm = r07;
        WHEN r07 =>           -- rg_h, clmp_h[], busy_h active (adclks)
            rosm = r08;
        WHEN r08 =>           -- rg_h, clmp_h[], busy_h active (adclks)
            rosm = r09;
        WHEN r09 =>           -- srgl_l, srgl_h, busy_h active
            rosm = r10;
        WHEN r10 =>           -- srgl_l, srgl_h, busy_h active
            rosm = r11;
        WHEN r11 =>           -- srgl_l, srgl_h, busy_h active
            rosm = r12;
        WHEN r12 =>           -- srgl_l, srgl_h, cnten_h, busy_h active
            IF cnttc_h THEN rosm = r13;
            ELSE rosm = r05;
            END IF;

        WHEN r13 =>           -- clmp_h[], busy_h active
            IF sync_h THEN
                dv_h.r = VCC;
                rosm = r00;
            END IF;
    END CASE;

END;
```

```
%=========================================================================
A/D Controller

Kensal Corporation
Phase II Camera Head

Author: Ken Crocker
Date:   30 Sep 1996
File:   ADCTRL.TDF
Rev:    1.00
=========================================================================%
TITLE "Phase II Camera Prototype - A/D Controller";

SUBDESIGN adctrl (
    gtcclk_r            : INPUT;        -- 40MHz clock from TC217 Controller
    grlclk_r            : INPUT;        -- 8.3333MHz clock from RL4000P Controller
    gclr_l              : INPUT;        -- active low asynchronous system reset

    -- Interface with TC217 Controller
    tcbusy_h            : INPUT;
    tcblank_h           : INPUT;
    tcdv_h              : INPUT;

    -- Interface with RL4000P Controller
    rlbusy_h            : INPUT;
    rlblank_h           : INPUT;
    rldv_h              : INPUT;

    -- Interface with Link Controller
    busy_h              : OUTPUT;
    blank_h             : OUTPUT;
    ramreq_h            : INPUT;
    ramack_h            : OUTPUT;
    aderr_h             : OUTPUT;
    clraderr_l          : INPUT;

    -- Interface with A/D and FIFO
    tcadclk_r[3..1]     : OUTPUT;
    tcraoe_l[3..1]      : OUTPUT;
    tcra_h[12..10]      : OUTPUT;
    tcwclk_r            : OUTPUT;
    tcwen_l             : OUTPUT;

    rladclk_r[3..1]     : OUTPUT;
    rlraoe_l[3..1]      : OUTPUT;
    rlra_h[12..10]      : OUTPUT;
    rlwclk_r            : OUTPUT;
    rlwen_l             : OUTPUT;

    tcclksel_h          : OUTPUT;
    tfir_l              : INPUT;

    -- Interface with LUT RAM
    rce_l               : OUTPUT;

    -- Diagnostic Signals
    tcsyncerr_h         : OUTPUT;
    rlsyncerr_h         : OUTPUT;
    tcovflw_h           : OUTPUT;
    rlovflw_h           : OUTPUT;
)

VARIABLE
    tcclk_r             : NODE;
```

```
rlclk_r              : NODE;
clr_l                : NODE;

tcadclktri_r[3..1]   : TRI;
tcraoetri_l[3..1]    : TRI;
tcratri_h[12..10]    : TRI;
tcwclktri_r          : TRI;
tcwentri_l           : TRI;
tcraoutd_h[11..10]   : LCELL;
tcwenoutd_l          : DFF;

rlclksel_l           : LCELL;
rladclktri_r[3..1]   : TRI;
rlraoetri_l[3..1]    : TRI;
rlratri_h[12..10]    : TRI;
rlwclktri_r          : TRI;
rlwentri_l           : TRI;
rlraoutd_h[10]       : LCELL;
rlwenoutd_l          : DFF;

tcsyncerr_h          : SRFF;
rlsyncerr_h          : SRFF;
tcovflw_h            : SRFF;
rlovflw_h            : SRFF;

tcbusysync_h         : DFF;      -- synchronizing F/F's
rlbusysync_h         : DFF;
tcdonesync_h         : DFF;
rldonesync_h         : DFF;
ramreqsync_h         : DFF;
adSM                 : MACHINE OF BITS (
                          busy_h, rloe_l, tcoe_l, ramack_h, rce_l
                       ) WITH STATES (
                          ad00     = B"01101",
                          ad01     = B"11000",
                          ad02     = B"01000",
                          ad03     = B"10100",
                          ad04     = B"00100",
                          ad05     = B"01110"
                       );

tcSM                 : MACHINE OF BITS (
                          tcadclkout_r[3..1], tcraoeout_l[3..1], tcraout_h[11..10],
                          tcwenout_l, tccntclr_h, tccntinc_h
                       ) WITH STATES (
                          tc00     = B"10011100110",
                          tc01     = B"00111100110",
                          tc02     = B"01011100110",

                          tc03     = B"10011100110",
                          tc04     = B"00111100110",
                          tc05     = B"01011100110",

                          tc06     = B"10011100100",
                          tc07     = B"00111100100",
                          tc08     = B"01011100101",

                          tc09     = B"10011000010",
                          tc10     = B"00110101010",
                          tc11     = B"01001110010",

                          tc12     = B"10011000000",
                          tc13     = B"00110101000",
                          tc14     = B"01001110001",
```

```
                    tc15     = B"10011100110",
                    tc16     = B"00111100110",
                    tc17     = B"01011100110"
                );

    rlSM            : MACHINE OF BITS (
                    rladclkout_r[3..1], rlraoeout_l[3..1], rlraout_h[10],
                    rlwenout_l, rlcntclr_h, rlcntinc_h
                ) WITH STATES (
                    rl00     = B"1111110110",
                    rl01     = B"0001110110",

                    rl02     = B"1111110110",
                    rl03     = B"0001110110",

                    rl04     = B"1111110100",
                    rl05     = B"0001110101",

                    rl06     = B"1111100010",
                    rl07     = B"0001011010",

                    rl08     = B"1111100000",
                    rl09     = B"0001011001",

                    rl10     = B"1111110110",
                    rl11     = B"0001110110"
                );

    tccnt_h[3..0]       : DFF;
    tccnttc_h           : LCELL;
    rlcnt_h[3..0]       : DFF;
    rlcnttc_h           : LCELL;

BEGIN
    tcclk_r             = GLOBAL (gtcclk_r);
    rlclk_r             = GLOBAL (grlclk_r);
    clr_l               = GLOBAL (gclr_l);

    tcsyncerr_h.clk     = tcclk_r;
    !tcsyncerr_h.clrn   = !gclr_l # !clraderr_l;
    tcsyncerr_h.r       = GND;

    rlsyncerr_h.clk     = rlclk_r;
    !rlsyncerr_h.clrn   = !gclr_l # !clraderr_l;
    rlsyncerr_h.r       = GND;

    tcovflw_h.clk       = tcclk_r;
    !tcovflw_h.clrn     = !gclr_l # !clraderr_l;
    tcovflw_h.s         = tfir_l & !tcwenout_l;
    tcovflw_h.r         = GND;

    rlovflw_h.clk       = rlclk_r;
    !rlovflw_h.clrn     = !gclr_l # !clraderr_l;
    rlovflw_h.s         = tfir_l & !rlwenout_l;
    rlovflw_h.r         = GND;

    aderr_h             = tcsyncerr_h # rlsyncerr_h # tcovflw_h # rlovflw_h;

    -- Tri-state Output Enables
    -- 'tcclksel_h' is active when adSM is in state ad01 or ad02 (ie. TC controller owns the bus).
    -- When active, signals tcadclk_r[3..1], tcwclk_r, and tcwen_l will be low Z.
    tcclksel_h          = !tcoe_l;
    -- The complimentary signals rladclk_r[3..1], rlwclk_r, and rlwen_l will be low Z when
```

```
-- 'rlclksel_l' is active, which occurs when the adSM is in state ad03, ad04 or ad05.
!rlclksel_l          = !rloe_l # ramack_h;        -- LCELL

-- As for the SRAM address and output enable signals, 'tcraoe_l[]' and 'tcra_h[]' will only
-- be low-Z when 'tcoe_l' is active. Similarly, the complimentary signals 'rlraoe_l[]' and
-- 'rlra_h[]' will only be low-Z when 'rloe_l' is active. These signals will be floating in
-- states ad00 and ad05.

-- Tri-state Outputs
tcadclktri_r[].in    = tcadclkout_r[];
tcadclktri_r[].oe    = !tcoe_l;
tcadclk_r[]          = tcadclktri_r[].out;
tcraoetri_l[].in     = tcraoeout_l[];
tcraoetri_l[].oe     = !tcoe_l;
tcraoe_l[]           = tcraoetri_l[].out;
tcraoutd_h[]         = tcraout_h[];
tcratri_h[].in       = (B"0", tcraoutd_h[]);
tcratri_h[].oe       = !tcoe_l;
tcra_h[]             = tcratri_h[].out;
tcwclktri_r.in       = gtcclk_r;
tcwclktri_r.oe       = !tcoe_l;
tcwclk_r             = tcwclktri_r.out;
tcwenoutd_l.clk      = !gtcclk_r;
!tcwenoutd_l.prn     = !clr_l;
!tcwenoutd_l.d       = !tcwenout_l;
tcwentri_l.in        = tcwenoutd_l.q;
tcwentri_l.oe        = !tcoe_l;
tcwen_l              = tcwentri_l.out;

rladclktri_r[].in    = rladclkout_r[];
rladclktri_r[].oe    = !rlclksel_l;
rladclk_r[]          = rladclktri_r[].out;
rlraoetri_l[].in     = rlraoeout_l[];
rlraoetri_l[].oe     = !rloe_l;
rlraoe_l[]           = rlraoetri_l[].out;
rlraoutd_h[]         = rlraout_h[];
rlratri_h[].in       = (B"10", rlraoutd_h[]);
rlratri_h[].oe       = !rloe_l;
rlra_h[]             = rlratri_h[].out;
rlwclktri_r.in       = grlclk_r;
rlwclktri_r.oe       = !rlclksel_l;
rlwclk_r             = rlwclktri_r.out;
rlwenoutd_l.clk      = !grlclk_r;
!rlwenoutd_l.prn     = !clr_l;
!rlwenoutd_l.d       = !rlwenout_l;
rlwentri_l.in        = rlwenoutd_l.q;
rlwentri_l.oe        = !rlclksel_l;
rlwen_l              = rlwentri_l.out;

-- TC217 Pipeline Counter
tccnt_h[].clk        = tcclk_r;
!tccnt_h[].clrn      = !clr_l;
IF tccntclr_h THEN
    tccnt_h[].d = 0;
ELSIF tccntinc_h THEN
    tccnt_h[].d = tccnt_h[].q + 1;
ELSE
    tccnt_h[].d = tccnt_h[].q;
END IF;
tccnttc_h    = (tccnt_h[].q == 11);       -- implemented as an LCELL

-- RL4000P Pipeline Counter
rlcnt_h[].clk        = rlclk_r;
!rlcnt_h[].clrn      = !clr_l;
```

```
IF rlcntclr_h THEN
    rlcnt_h[].d = 0;
ELSIF rlcntinc_h THEN
    rlcnt_h[].d = rlcnt_h[].q + 1;
ELSE
    rlcnt_h[].d = rlcnt_h[].q;
END IF;
rlcnttc_h   = (rlcnt_h[].q == 11);      -- implemented as an LCELL

-- Blank signal to Link Controller
blank_h = tcblank_h # rlblank_h;

%================================================================
Output Enable State Machine

This state machine serves as an arbiter of the TC217, RL4000P and
SRAM signal busses. It switches state when a request is made internally
from either the tcSM or the rlSM, or an external request comes in via
'ramreq_h'.
================================================================%

tcbusysync_h.clk     = tcclk_r;
!tcbusysync_h.clrn   = !clr_l;
tcbusysync_h.d       = tcbusy_h;

rlbusysync_h.clk     = tcclk_r;
!rlbusysync_h.clrn   = !clr_l;
rlbusysync_h.d       = rlbusy_h;

tcdonesync_h.clk     = tcclk_r;
!tcdonesync_h.clrn   = !clr_l;
tcdonesync_h.d       = (tcSM == tc00);

rldonesync_h.clk     = tcclk_r;
!rldonesync_h.clrn   = !clr_l;
rldonesync_h.d       = (rlSM == rl00);

ramreqsync_h.clk     = tcclk_r;
!ramreqsync_h.clrn   = !clr_l;
ramreqsync_h.d       = ramreq_h;

adSM.clk    = tcclk_r;
adSM.reset  = !clr_l;

CASE adSM IS
    -- Power on state; nobody owns the bus
    WHEN ad00 =>          -- <nothing> active
        IF ramreqsync_h THEN adSM = ad05;
        ELSIF tcbusysync_h THEN adSM = ad01;
        ELSIF rlbusysync_h THEN adSM = ad03;
        ELSE adSM = ad04;
        END IF;

    -- TC217 Controller owns the bus
    WHEN ad01 =>          -- busy_h, tcoe_l, rce_l active
        IF tcdonesync_h THEN adSM = ad02;
        END IF;
    WHEN ad02 =>          -- tcoe_l, rce_l active
        IF tcbusysync_h THEN adSM = ad01;
        ELSIF rlbusysync_h # ramreqsync_h THEN adSM = ad00;
        END IF;

    -- RL4000P Controller owns the bus
    WHEN ad03 =>          -- busy_h, rloe_l, rce_l active
        IF rldonesync_h THEN adSM = ad04;
```

```
        END IF;
    -- Default state to go to. RL4000P Controller owns bus but is inactive.
    WHEN ad04 =>        -- rloe_l, rce_l active
        IF rlbusysync_h THEN adSM = ad03;
        ELSIF tcbusysync_h # ramreqsync_h THEN adSM = ad00;
        END IF;

    -- The Link Controller owns the bus
    WHEN ad05 =>        -- ramack_h, rce_l active
        IF !ramreqsync_h THEN adSM = ad00;
        END IF;
END CASE;

%=====================================================================
TC217 State Machine
=====================================================================%
tcSM.clk    = tcclk_r;
tcSM.reset  = !clr_l;

CASE tcSM IS
    -- Wait for tcbusy_h and sync up to it...
    WHEN tc00 =>                -- tcadclk_r1, tccntclr_h active
        IF tcbusy_h THEN tcSM = tc03;
        ELSE tcSM = tc01;
        END IF;
    WHEN tc01 =>                -- tcadclk_r2, tccntclr_h active
        IF tcbusy_h THEN tcSM = tc03;
        ELSE tcSM = tc02;
        END IF;
    WHEN tc02 =>                -- tcadclk_r3, tccntclr_h active
        IF tcbusy_h THEN tcSM = tc03;
        ELSE tcSM = tc00;
        END IF;

    -- Wait for tcdv_h and check synchronization...
    WHEN tc03 =>                -- tcadclk_r1, tccntclr_h active
        IF !tcbusy_h THEN tcSM = tc01;
        ELSIF tcdv_h THEN tcSM = tc06; tcsyncerr_h.s = VCC;
        ELSE tcSM = tc04;
        END IF;
    WHEN tc04 =>                -- tcadclk_r2, tccntclr_h active
        IF !tcbusy_h THEN tcSM = tc02;
        ELSIF tcdv_h THEN tcSM = tc06; tcsyncerr_h.s = VCC;
        ELSE tcSM = tc05;
        END IF;
    WHEN tc05 =>                -- tcadclk_r3, tccntclr_h active
        IF !tcbusy_h THEN tcSM = tc00;
        ELSIF tcdv_h THEN tcSM = tc06;
        ELSE tcSM = tc03;
        END IF;

    -- Count off 12 A/D clocks before writing data to FIFO
    WHEN tc06 =>                -- tcadclk_r1 active
        tcSM = tc07;
    WHEN tc07 =>                -- tcadclk_r2 active
        tcSM = tc08;
    WHEN tc08 =>                -- tcadclk_r3, tccntinc_h active
        IF tccnttc_h THEN tcSM = tc09;
        ELSE tcSM = tc06;
        END IF;

    -- Data valid at A/D *AND* FIFO
    WHEN tc09 =>                -- tcadclk_r1, tcoel_l, tcraout_h[]=B"00", tcwen_l, tccntclr_h
active
```

```
        IF !tcdv_h THEN tcsyncerr_h.s = VCC;
        END IF;
        tcSM = tc10;
    WHEN tc10 =>                -- tcadclk_r2, tcoe2_l, tcraout_h[]=B"01", tcwen_l, tccntclr_h
active
        IF !tcdv_h THEN tcsyncerr_h.s = VCC;
        END IF;
        tcSM = tc11;
    WHEN tc11 =>                -- tcadclk_r3, tcoe3_l, tcraout_h[]=B"10", tcwen_l, tccntclr_h
active
        IF tcdv_h THEN tcSM = tc09;
        ELSE tcSM = tc12;
        END IF;

    -- Data valid at FIFO for 12 more clocks
    WHEN tc12 =>                -- tcadclk_r1, tcoe1_l, tcraout_h[]=B"00", tcwen_l active
        tcSM = tc13;
    WHEN tc13 =>                -- tcadclk_r2, tcoe2_l, tcraout_h[]=B"01", tcwen_l active
        tcSM = tc14;
    WHEN tc14 =>                -- tcadclk_r3, tcoe3_l, tcraout_h[]=B"10", tcwen_l, tccntinc_h
active
        IF !tccnttc_h THEN tcSM = tc12;
        ELSE tcSM = tc15;
        END IF;

    -- Wait 3 clocks for data to reach the output of the transmit FIFO (delays falling edge of
busy_h)
    WHEN tc15 =>                -- tcadclk_r1, tcraout_h[]=B"00", tccntclr_h active
        tcSM = tc16;
    WHEN tc16 =>                -- tcadclk_r2, tcraout_h[]=B"00", tccntclr_h active
        tcSM = tc17;
    WHEN tc17 =>                -- tcadclk_r3, tcraout_h[]=B"00", tccntclr_h active
        IF !tcbusy_h THEN tcSM = tc00;
        ELSE tcSM = tc03;
        END IF;

END CASE;

%==================================================================
RL4000P State Machine
==================================================================%
rlSM.clk        = rlclk_r;
rlSM.reset      = !clr_l;

CASE rlSM IS
    -- Wait for rlbusy_h and sync up to it...
    WHEN rl00 =>                -- rladclk_r1, rladclk_r2, rladclk_r3, rlcntclr_h active
        IF rlbusy_h THEN rlSM = rl02;
        ELSE rlSM = rl01;
        END IF;
    WHEN rl01 =>                -- rlcntclr_h active
        IF rlbusy_h THEN rlSM = rl02;
        ELSE rlSM = rl00;
        END IF;

    -- Wait for rldv_h and check synchronization...
    WHEN rl02 =>                -- rladclk_r1, rladclk_r2, rladclk_r3, rlcntclr_h active
        IF !rlbusy_h THEN rlSM = rl01;
        ELSIF rldv_h THEN rlsyncerr_h.s = VCC; rlSM = rl04;
        ELSE rlSM = rl03;
        END IF;
    WHEN rl03 =>                -- rlcntclr_h active
        IF !rlbusy_h THEN rlSM = rl00;
        ELSIF rldv_h THEN rlSM = rl04;
```

```
            ELSE r1SM = r102;
            END IF;

        -- Count off 12 A/D clocks before writing data to FIFO
        WHEN r104 =>                    -- rladclk_r1, rladclk_r2, rladclk_r3 active
            r1SM = r105;
        WHEN r105 =>                    -- rlcntinc_h active
            IF !rlcnttc_h THEN r1SM = r104;
            ELSE r1SM = r106;
            END IF;

        -- Data valid at A/D *AND* FIFO
        WHEN r106 =>                    -- rladclk_r1, rladclk_r2, rladclk_r3, rloe1_l, rlraout_h[10]=0,
rlwen_l, rlcntclr_h active
            rlsyncerr_h.s = !rldv_h;
            r1SM = r107;
        WHEN r107 =>                    -- rloe2_l, rlraout_h[10]=1, rlwen_l, rlcntclr_h active
            IF rldv_h THEN r1SM = r106;
            ELSE r1SM = r108;
            END IF;

        -- Data valid at FIFO for 12 more clocks
        WHEN r108 =>                    -- rladclk_r1, rladclk_r2, rladclk_r3, rloe1_l, rlraout_h[10]=0,
rlwen_l active
            r1SM = r109;
        WHEN r109 =>                    -- rloe2_l, rlwen_l, rlraout_h[10]=1, rlcntinc_h active
            IF !rlcnttc_h THEN r1SM = r108;
            ELSE r1SM = r110;
            END IF;

        -- Allow two clocks for data to get to the output of the FIFO (delays falling edge of busy_h)
        WHEN r110 =>                    -- rladclk_r1, rladclk_r2, rladclk_r3, rlcntclr_h active
            r1SM = r111;
        WHEN r111 =>                    -- rlcntclr_h active
            IF !rlbusy_h THEN r1SM = r100;
            ELSE r1SM = r102;
            END IF;

    END CASE;
END;
```

```
%=========================================================================
Anti-blooming Gate Controller

Kensal Corporation
Phase II Camera Prototype

Author: Ken Crocker
Date:   4 Mar 96
File:   ABGC.TDF
Rev:    1.10
=========================================================================%
TITLE "Phase II Camera Prototype - Anti-Blooming Gate Controller";

SUBDESIGN abgc
(
    gclk_r          : INPUT;
    gclr_l          : INPUT;

    abgcont_h       : INPUT;
    abgpulse_h      : INPUT;

    abgin_l         : OUTPUT;       -- active low ABG pulse
    busy_h          : OUTPUT;
    hzsync_h        : OUTPUT;
)

VARIABLE
    clk_r           : NODE;
    clr_l           : NODE;

    hzsync_h        : DFF;

    seq_h[3..0]     : DFF;
    sync_h          : NODE;
    seqend_h        : NODE;

    fastabg_h       : SRFF;

    cnt_h[3..0]     : DFF;
    cntend_h        : NODE;
    cntclr_h        : NODE;
    cntld_h         : NODE;
    cnten_h         : NODE;

    abgsm           : MACHINE WITH STATES (a00, a01, a02, a03);

BEGIN
    clk_r   = GLOBAL (gclk_r);
    clr_l   = GLOBAL (gclr_l);

    busy_h = !a00;

    -- 'hzsync_h' provides synchronization to the HZTMG controller during line
    -- transfers.  It goes active for one clock period in the middle of the
    -- second ABG pulse.
    hzsync_h.clk = clk_r;
    hzsync_h.clrn = clr_l;

%=========================================================================
    Only one of the following statements should be commented out.
        First statement commented   => disable ABG.
        Second statement commented  => enable ABG.
=========================================================================%
    abgin_l = !a02;
```

```
--  abgin_l = VCC;

    fastabg_h.clk = clk_r;              -- when 0, we're doing continuous ABGs
    fastabg_h.clrn = clr_l;          -- when 1, we're doing a burst of fast ABGs

    -- seq_h[] is used to derrive the frequency of the ABG pulses from the 20 MHz
    -- clk_r input.  Two speeds are available: 1.25 MHz and 1.00 MHz (fast and slow).
    -- The speed selection depends on the setting of 'fastabg_h', which in turn
    -- is determined by whether 'abgcont_h' or 'abgpulse_h' was used to start the
    -- sequence.
    seq_h[].clk = clk_r;
    seq_h[].clrn = clr_l;
    IF seqend_h # sync_h THEN
        seq_h[].d = 0;
    ELSE
        seq_h[].d = seq_h[].q + 1;
    END IF;

    -- sequence end decoding
    seqend_h = (fastabg_h & (seq_h[].q == 7)) # (!fastabg_h & (seq_h[].q == 9));

    -- cnt_h[] is used to count the number of ABG pulses to issue when 'abgpulse_h'
    -- is activated.  Currently, 9 ABG pulses will be issued every time
    -- 'abgpulse_h' goes active.
    cnt_h[].clk = clk_r;
    cnt_h[].clrn = clr_l;
    IF cntclr_h THEN                 -- 'cntclr_h' clears cnt_h[] to 0
        cnt_h[].d = 0;
    ELSIF cntld_h THEN               -- 'cntld_h' sets cnt_h[] at it's terminal cnt_h
        cnt_h[].d = 9;
    ELSIF cnten_h & !cntend_h THEN   -- 'cnten_h' increments cnt_h[] if not at tc
        cnt_h[].d = cnt_h[].q + 1;
    ELSE
        cnt_h[].d = cnt_h[].q;
    END IF;

    -- count end decoding
    cntend_h = (cnt_h[].q == 9);        --the 9th clock on a 4-bit binary counter

    abgsm.clk   = clk_r;
    abgsm.reset = !clr_l;

%===========================================================================
    Anti-Blooming Gate State Machine
===========================================================================%

    CASE (abgsm) IS
        WHEN a00 =>
            sync_h = VCC;
            cntld_h = VCC;              --cntclr_h overrides this when active
            IF abgcont_h THEN
                fastabg_h.r = VCC;
                abgsm = a01;
            ELSIF abgpulse_h THEN
                fastabg_h.s = VCC;
                cntclr_h = VCC;
                abgsm = a02;
            END IF;

        WHEN a01 =>                     -- busy_h active
            IF seqend_h THEN
                abgsm = a02;
            END IF;

        WHEN a02 =>                     -- busy_h active; abgin_l active (low)
```

```
            IF fastabg_h & cnt_h[].q == 1 & seq_h[].q == 3 THEN
                hzsync_h.d = VCC;
            END IF;
            IF seqend_h THEN
                cnten_h = VCC;
                abgsm = a03;
            END IF;

        WHEN a03 =>                                   -- busy_h active
            IF seqend_h THEN
                IF !cntend_h THEN
                    abgsm = a02;
                ELSIF abgcont_h THEN
                    fastabg_h.r = VCC;
                    abgsm = a02;
                ELSIF abgpulse_h THEN
                    fastabg_h.s = VCC;
                    cntclr_h = VCC;
                    abgsm = a02;
                ELSE
                    abgsm = a00;
                END IF;
            END IF;
    END CASE;

END;
```

```
%=========================================================================
CCD State Machine Controller

Kensal Corporation
Phase II Camera Prototype

Author: Ken Crocker
Date:   6 Nov 1996
File:   CCDSMC.TDF
Rev:    1.00
=========================================================================%
TITLE "Phase II Camera Prototype - CCD State Machine Controller";

SUBDESIGN ccdsmc
(
    clk_r              : INPUT;
    clr_l              : INPUT;

    go_h               : INPUT;
    even_h             : INPUT;

    xfrbusy_h          : INPUT;        -- Input from PARXFR controller
    hzbusy_h           : INPUT;        -- Inputs from HZTMG controller
    readbusy_h         : INPUT;        -- Input from READOUT controller

    sync_h             : INPUT;
    vcnt243_h          : INPUT;

    xfrodd_h           : OUTPUT;       -- Outputs to PARXFR controller
    xfreven_h          : OUTPUT;

    line1_h            : OUTPUT;
    line2_h            : OUTPUT;

    doread_h           : OUTPUT;       -- Outputs to READOUT controller
    darkline_h         : OUTPUT;

    vcntclr_h          : OUTPUT;
    vcntena_h          : OUTPUT;

    busy_h             : OUTPUT;
)

VARIABLE
    -- Output Redeclarations
    xfrodd_h           : SRFF;         -- Outputs to PARXFR controller
    xfreven_h          : SRFF;

    line1_h            : SRFF;
    line2_h            : SRFF;

    doread_h           : SRFF;         -- Outputs to READOUT controller

    vcntclr_h          : DFF;
    vcntena_h          : DFF;

    -- Internal nodes
    readbusyd_h        : DFF;          -- resynchronizing ff

    darkline_h         : SRFF;         -- active if we're reading first two lines

    ccdsm              : MACHINE OF BITS (
                           busy_h
                       ) WITH STATES (
```

```
                        c00 = B"0",
                        c01 = B"1",
                        c02 = B"1",
                        c03 = B"1",
                        c04 = B"1",
                        c05 = B"1"
                    );

BEGIN
    readbusyd_h.clk = clk_r;
    readbusyd_h.d = readbusy_h;

    xfrodd_h.clk = clk_r;
    xfrodd_h.r = xfrbusy_h;
    xfreven_h.clk = clk_r;
    xfreven_h.r = xfrbusy_h;

    line1_h.clk = clk_r;
    line1_h.r = hzbusy_h;
    line2_h.clk = clk_r;
    line2_h.r = hzbusy_h;

    doread_h.clk = clk_r;
    doread_h.r = readbusyd_h;

    vcntclr_h.clk = clk_r;
    vcntena_h.clk = clk_r;

    -- 'darkline_h'
    -- The first two lines read out of a TC217 are dark lines. If 'allpixels_h'
    -- is active, these lines are digitized. If 'allpixels_h' is inactive, the
    -- two lines are read out but not digitized with the result that only
    -- lines with active pixels are digitized and sent to the Frame Capture Board.
    darkline_h.clk  = clk_r;
    !darkline_h.prn = !clr_l;

    ccdsm.clk   = clk_r;
    ccdsm.reset = !clr_l;

    %=============================================================================
    CCD State Machine
    =============================================================================%

    CASE (ccdsm) IS
        -- waiting for 'go_h' and 'sync_h' to be active
        WHEN c00 =>          -- (xfreven_h.s or xfrodd_h.s) may be active
            darkline_h.s = VCC;
            IF go_h & sync_h THEN
                ccdsm = c01;
                IF even_h THEN xfreven_h.s = VCC;
                ELSE xfrodd_h.s = VCC;
                END IF;
            END IF;

        -- waiting for IA -> SA transfer (odd or even field) to conclude
        WHEN c01 =>          -- busy_h (xfreven_h or xfrodd_h) active
            IF (!xfreven_h & !xfrodd_h & !xfrbusy_h) THEN
                vcntclr_h.d = VCC;
                line2_h.s = VCC;
                ccdsm = c02;
            END IF;

        -- waiting for line 2 horizontal transfer to conclude
        WHEN c02 =>          -- busy_h active; (doread_h.s) may be active
            IF (!line2_h & !hzbusy_h) THEN
```

```
                doread_h.s = VCC;
                ccdsm = c03;
            END IF;
        -- waiting for line 2 readout to conclude
        WHEN c03 =>         -- busy_h active; (doread_h, line1_h.s) may be active
            IF (!doread_h & !readbusyd_h) THEN
                line1_h.s = VCC;
                ccdsm = c04;
            END IF;
        -- waiting for line 1 horizontal transfer to conclude
        WHEN c04 =>         -- busy_h active; (doread_h.s) may be active
            IF (!line1_h & !hzbusy_h) THEN
                doread_h.s = VCC;
                ccdsm = c05;
            END IF;
        -- waiting for line 1 readout to conclude
        WHEN c05 =>         -- busy_h active
            IF (!doread_h & !readbusyd_h) THEN
                darkline_h.r = VCC;
                IF (!vcnt243_h) THEN
                    line2_h.s = VCC;
                    vcntena_h.d = VCC;
                    ccdsm = c02;
                ELSE
                    ccdsm = c00;
                END IF;
            END IF;
    END CASE;

END;
```

```
%==========================================================================
Horizontal Timing Controller

Kensal Corporation
Phase II Camera Prototype

Author: Ken Crocker
Date:   6 Mar 96
File:   HZTMGC.TDF
Rev:    1.20
==========================================================================%
TITLE "Phase II Camera Prototype - Horizontal Timing Controller";

SUBDESIGN hztmgc
(
    clk_r           : INPUT;
    clr_l           : INPUT;

    line1_h         : INPUT;    -- activates SAG1 during HT
    line2_h         : INPUT;    -- activates SAG2 during HT

    sag_h[2..1]     : OUTPUT;
    srgon_h         : OUTPUT;
    trg_h           : OUTPUT;

    busy_h          : OUTPUT;
)

VARIABLE
    sag_h[2..1]     : SRFF;
    srgon_h         : SRFF;
    trg_h           : SRFF;

    usesag1_h       : SRFF;

    seq_h[3..0]     : DFF;

    sync_h          : NODE;
    seqend_h        : NODE;

-- State Definitions
    htsm            : MACHINE WITH STATES (
                            h00, h01, h02, h03, h04, h05, h06, h08, h09, h10
                        );

BEGIN
    sag_h[].clk = clk_r;
    sag_h[].clrn = clr_l;
    srgon_h.clk = clk_r;
    srgon_h.clrn = clr_l;
    trg_h.clk = clk_r;
    trg_h.clrn = clr_l;

    seqend_h = (seq_h[].q == 8);
    seq_h[].clk = clk_r;
    seq_h[].clrn = clr_l;
    IF (seqend_h # sync_h) THEN
        seq_h[].d = 0;
    ELSE
        seq_h[].d = seq_h[].q + 1;
    END IF;

    usesag1_h.clk = clk_r;
    usesag1_h.clrn = clr_l;
```

```
busy_h = !h00;

htsm.clk = clk_r;
htsm.reset = !clr_l;

%=========================================================================
    Horizontal Timing State Machine
=========================================================================%

    CASE (htsm) IS
        WHEN h00 =>          -- <nothing> active
            srgon_h.r = VCC;
            sync_h = VCC;
            IF line1_h THEN
                usesag1_h.s = VCC;
                htsm = h08;
            ELSIF line2_h THEN
                usesag1_h.r = VCC;
                htsm = h09;
            END IF;

        WHEN h01 =>          -- sag_h2, trg_h active
            IF seqend_h THEN
                sag_h[].r = b"11";
                srgon_h.s = VCC;
                trg_h.r = VCC;
                htsm = h02;
            END IF;

        WHEN h02 =>          -- srg_h[] active
            IF seqend_h THEN
                srgon_h.r = VCC;
                trg_h.s = VCC;
                IF usesag1_h THEN
                    sag_h1.s = VCC;
                ELSE
                    sag_h2.s = VCC;
                END IF;
                htsm = h03;
            END IF;

        WHEN h03 =>
            IF seqend_h THEN
                sag_h[].r = b"11";
                srgon_h.s = VCC;
                trg_h.r = VCC;
                htsm = h04;
            END IF;

        WHEN h04 =>
            IF seqend_h THEN
                srgon_h.r = VCC;
                trg_h.s = VCC;
                IF usesag1_h THEN
                    sag_h1.s = VCC;
                ELSE
                    sag_h2.s = VCC;
                END IF;
                htsm = h05;
            END IF;

        WHEN h05 =>
            IF seqend_h THEN
                sag_h[].r = b"11";
```

```
                srgon_h.s = VCC;
                trg_h.r = VCC;
                htsm = h06;
            END IF;

        WHEN h06 =>
            IF seqend_h THEN
                srgon_h.r = VCC;
                htsm = h10;
            END IF;

        WHEN h08 =>     -- will use sag_h1 on next seqend_h
            IF seqend_h THEN
                sag_h1.s = VCC;
                trg_h.s = VCC;
                htsm = h01;
            END IF;

        WHEN h09 =>     -- will use sag_h2 on next seqend_h
            IF seqend_h THEN
                sag_h2.s = VCC;
                trg_h.s = VCC;
                htsm = h01;
            END IF;

        WHEN h10 =>
            IF seqend_h THEN
                htsm = h00;
            END IF;
    END CASE;
END;
```

```
INCLUDE "CCDSMC.INC";
INCLUDE "HZTMGC.INC";
INCLUDE "PARXFRC.INC";
INCLUDE "READOUTC.INC";

SUBDESIGN main
(
    gclk80_r        : INPUT;        -- 80 MHz clock
    gclr_l          : INPUT;        -- active low asynchronous system reset
--  oe_l            : INPUT;        -- reserved OE pin
    simulating_h    : INPUT;        -- if VCC, then we're simulating

    -- Interface with A/D Controller
    tcgo_h          : INPUT;
    tceven_h        : INPUT;
    allpixels_h     : INPUT;
    tcclk_r         : OUTPUT;       -- 40 MHz clock to A/D Controller
    tcbusy_h        : OUTPUT;       -- active whenever CCD Controller is busy
    tcblank_h       : OUTPUT;       -- active during IA -> SA transfer
    tcdv_h          : OUTPUT;       -- active whenever A/D input is valid

    -- Interface with TC217
    midsel_h        : OUTPUT;
    iagin_h         : OUTPUT;
    tmg_h           : OUTPUT;
    sag_h[2..1]     : OUTPUT;
    srga_h[3..1]    : OUTPUT;
    srgb_h[3..1]    : OUTPUT;
    abgin_l         : OUTPUT;
    trg_h           : OUTPUT;

    -- Interface with clamp and sample circuit
    clamp_h[3..1]   : OUTPUT;
    sample_h[3..1]  : OUTPUT;
    dis_h           : OUTPUT;       -- Elantec EL2071 disable

    -- Diagnostic signals
    vsync_h         : OUTPUT;
    hsync_h         : OUTPUT;
    sync_h          : OUTPUT;
)

VARIABLE
    clk80_r         : NODE;
    clr_l           : NODE;

    -- Output Redeclarations
    clk40_r         : TFF;
    clk20_r         : TFF;
    tcclk_r         : DFF;

    midsel_h        : DFF;
    iagin_h         : DFF;
    tmg_h           : DFF;
    sag_h[2..1]     : DFF;
    trg_h           : DFF;

    synccnt_h[1..0] : DFF;
    sync_h          : DFF;

    vcnt_h[7..0]    : DFF;
    vcnt243_h       : DFF;
    vcntclr_h       : LCELL;
    vcntena_h       : LCELL;
```

```
    vsync_h         : DFF;
    hsync_h         : DFF;

    ccdsm           : ccdsmc;
    tcgoin_h        : DFF;          -- F/F is in IOCELL
    line1_h         : NODE;
    line2_h         : NODE;
    doread_h        : NODE;
    darkline_h      : NODE;

    hztmg           : hztmgc;
    srgon_h         : NODE;
    hzbusy_h        : NODE;

    parxfr          : parxfrc;
    xfrodd_h        : NODE;
    xfreven_h       : NODE;
    xfrbusy_h       : NODE;

    readout         : readoutc;
    srg_h[3..1]     : NODE;
    readbusy_h      : NODE;

BEGIN
    clk80_r         = GLOBAL (gclk80_r);
    clr_l           = GLOBAL (gclr_l);

    -- 40 and 20 MHz Clock Generation
    clk40_r.clk = clk80_r;
    clk40_r.t   = VCC;
    clk20_r.clk = clk80_r;
    clk20_r.t   = !clk40_r.q;

    -- Interface to A/D Controller (partial)
    -- 'tcclk_r' is a copy in frequency and phase of 'clk40out_r'.
    -- See Readout Controller for 'tcdv_h' output.
    -- See CCD State Machine Controller for 'tcbusy_h' output.
    tcclk_r.clk     = clk80_r;
    tcclk_r.d       = !clk40_r.q;
    tcblank_h       = xfreven_h # xfrodd_h # xfrbusy_h;

    -- Synchronization Generator
    -- In order to minimize the number of signal lines between the TC217 and A/D Controllers
    -- we use 'tcbusy_h' to force synchronization of the tri-phase A/D clocks. The A/D
    -- Controller then verifies sync when 'tcdv_h' goes active. Therefore, there must be
    -- modulo 3 'tcclk_r' clocks between 'tcbusy_h' and 'tcdv_h'. 'synccnt_h[]' is a free
    -- running modulo 3 counter. 'sync_h' will go active once every three cycles.
    -- 'sync_h.q' is used as follows: When 'sync_h' goes active, then 'tcbusy_h' or 'tcdv_h'
    -- is allowed to go active on the next clock. The A/D Controller will then cause
    -- 'tcadclk_r1' (ie. the first of 3 phases) to go active on the next clock cycle.
    synccnt_h[].clk     = clk40_r;
    !synccnt_h[].clrn   = !clr_l;
    IF synccnt_h[] == 2 THEN
        synccnt_h[].d = 0;
    ELSE
        synccnt_h[].d = synccnt_h[].q + 1;
    END IF;
    sync_h.clk      = clk40_r;
    !sync_h.clrn    = !clr_l;
    sync_h.d        = (synccnt_h[] == 2);

    -- Vertical Counter
    vcnt_h[].clk    = clk20_r;
```

```
    vcnt_h[].clrn   = clr_l;
    IF vcntclr_h THEN
        vcnt_h[].d = 0;
    ELSIF vcntena_h THEN
        vcnt_h[].d = vcnt_h[].q + 1;
    ELSE
        vcnt_h[].d = vcnt_h[].q;
    END IF;
    vcnt243_h.clk = clk20_r;
    vcnt243_h.clrn = clr_l;
    IF simulating_h THEN
        vcnt243_h.d = (vcnt_h[].q == B"XXXXXX10");       -- (2 + 1) x 2 = 6 total - 2 dark = 4 active
    ELSE
        vcnt243_h.d = (vcnt_h[].q == B"1111XX11");       -- (243 + 1) x 2 = 488 total - 2 dark = 486
active
    END IF;
    vcntena_h = ccdsm.vcntena_h # parxfr.vcntena_h;     -- implemented as LCELL
    vcntclr_h = ccdsm.vcntclr_h # parxfr.vcntclr_h;     -- implemented as LCELL

    !abgin_l = GND;

    -- CCD State Machine Controller
    ccdsm.(clk_r, clr_l, go_h, even_h, xfrbusy_h, hzbusy_h, readbusy_h, sync_h, vcnt243_h)
        = (clk20_r, clr_l, tcgoin_h, tceven_h, xfrbusy_h, hzbusy_h, readbusy_h, sync_h, vcnt243_h.q);
    (xfrodd_h, xfreven_h, line1_h, line2_h, doread_h, darkline_h, tcbusy_h)
        = ccdsm.(xfrodd_h, xfreven_h, line1_h, line2_h, doread_h, darkline_h, busy_h);

    tcgoin_h.clk    = clk20_r;      -- Synchronizing F/F (in IOCELL)
    !tcgoin_h.clrn  = !clr_l;
    tcgoin_h.d      = tcgo_h;

    vsync_h.clk = clk20_r;
    vsync_h.clrn = clr_l;
    vsync_h.d = xfrodd_h # xfreven_h;

    hsync_h.clk = clk20_r;
    hsync_h.clrn = clr_l;
    hsync_h.d = line1_h # line2_h;

    -- Horizontal Timing Controller
    hztmg.(clk_r, clr_l, line1_h, line2_h)
        = (clk20_r, clr_l, line1_h, line2_h);
    (srgon_h, hzbusy_h) = hztmg.(srgon_h, busy_h);

    trg_h.clk   = clk40_r;
    !trg_h.clrn = !clr_l;
    trg_h.d     = hztmg.trg_h;

    -- Parallel Transfer Controller
    parxfr.(clk_r, clr_l, xfrodd_h, xfreven_h, vcnt243_h)
        = (clk20_r, clr_l, xfrodd_h, xfreven_h, vcnt243_h.q);
    xfrbusy_h = parxfr.busy_h;

    midsel_h.clk = clk20_r;
    midsel_h.clrn = clr_l;
    midsel_h.d = parxfr.midsel_h;

    iagin_h.clk = clk20_r;
    iagin_h.clrn = clr_l;
    iagin_h.d = parxfr.iagin_h;

    tmg_h.clk = clk20_r;
    tmg_h.clrn = clr_l;
    tmg_h.d = parxfr.tmg_h;
```

```
    sag_h[].clk = clk40_r;
    sag_h[].clrn = clr_l;
    sag_h[].d = parxfr.sag_h[] # hztmg.sag_h[];

    -- Readout Controller
    readout.(clk_r, clr_l, srgon_h, simulating_h, allpixels_h, darkline_h, go_h, sync_h)
        = (clk40_r, clr_l, srgon_h, simulating_h, allpixels_h, darkline_h, doread_h, sync_h);
    (tcdv_h, srg_h[], clamp_h[], sample_h[], readbusy_h)
        = readout.(dv_h, srg_h[], clamp_h[], sample_h[], busy_h);

    srga_h[] = srg_h[];
    srgb_h[] = srg_h[];

    -- Elantec output disable
    dis_h = !readbusy_h;
END;
```

```
%==========================================================================
Parallel Transfer Controller

Kensal Corporation
Phase II Camera Prototype

Author: Ken Crocker
Date:   4 Mar 96
File:   PARXFRC.TDF
Rev:    1.10
==========================================================================%
TITLE "Phase II Camera Prototype - Parallel Transfer Controller";

SUBDESIGN parxfrc
(
    clk_r           : INPUT;     -- 20 MHz clock (50ns period)
    clr_l           : INPUT;

    xfrodd_h        : INPUT;
    xfreven_h       : INPUT;

    vcnt243_h       : INPUT;

    midsel_h        : OUTPUT;
    iagin_h         : OUTPUT;
    tmg_h           : OUTPUT;
    sag_h[2..1]     : OUTPUT;

    vcntclr_h       : OUTPUT;
    vcntena_h       : OUTPUT;

    busy_h          : OUTPUT;
)

VARIABLE
    midsel_h        : SRFF;
    iagin_h         : SRFF;
    tmg_h           : SRFF;      -- signal is inverted by parallel dvr??
    sag_h1          : SRFF;
    sag_h2          : SRFF;

    vcntclr_h       : DFF;
    vcntena_h       : DFF;

    sag1sel_h       : SRFF;
    xfrdone_h       : SRFF;

    seq_h[1..0]     : DFF;

    sync_h          : NODE;
    seqend_h        : NODE;

    startup_h       : SRFF;

    ptsm            : MACHINE WITH STATES (
                        p00, p01, p02, p03, p04, p05,
                        p06, p07, p08, p09, p10, p11
                    );

BEGIN
    midsel_h.clk = clk_r;
    midsel_h.clrn = clr_l;
    iagin_h.clk = clk_r;
    iagin_h.clrn = clr_l;
```

```
tmg_h.clk = clk_r;
tmg_h.clrn = clr_l;
sag_h1.clk = clk_r;
sag_h1.clrn = clr_l;
sag_h2.clk = clk_r;
sag_h2.clrn = clr_l;

vcntclr_h.clk = clk_r;
vcntclr_h.clrn = clr_l;
vcntena_h.clk = clk_r;
vcntena_h.clrn = clr_l;

sag1sel_h.clk = clk_r;
sag1sel_h.clrn = clr_l;
xfrdone_h.clk = clk_r;
xfrdone_h.clrn = clr_l;

seq_h[].clk = clk_r;
seq_h[].clrn = clr_l;
IF (seqend_h # sync_h) THEN
    seq_h[].d = 0;
ELSE
    seq_h[].d = seq_h[].q + 1;
END IF;

seqend_h = (seq_h[].q == 2);

startup_h.clk = clk_r;
startup_h.clrn = clr_l;

busy_h = !p00;

ptsm.clk = clk_r;
ptsm.reset = !clr_l;

%=============================================================================
    Parallel Transfer State Machine
=============================================================================%

CASE (ptsm) IS
    WHEN p00 =>     -- tmg_h active
        xfrdone_h.r = VCC;
        sync_h = VCC;
        startup_h.s = VCC;
        IF xfrodd_h THEN
            midsel_h.s = VCC;
            iagin_h.r = VCC;
            vcntclr_h.d = VCC;
            ptsm = p01;
        ELSIF xfreven_h THEN
            midsel_h.s = VCC;
            iagin_h.s = VCC;
            tmg_h.r = VCC;
            vcntclr_h.d = VCC;
            ptsm = p04;
        ELSE
            midsel_h.r = VCC;
            tmg_h.s = VCC;
        END IF;

    WHEN p01 =>     -- midsel_h, tmg_h active
        IF seqend_h THEN
            ptsm = p02;
        END IF;
```

```
WHEN p02 =>     -- midsel_h, tmg_h active
    IF seqend_h THEN
        ptsm = p03;
    END IF;

WHEN p03 =>     -- midsel_h, tmg_h active
    IF seqend_h THEN
        iagin_h.s = VCC;
        tmg_h.r = VCC;
        sag_h2.s = !startup_h;
        ptsm = p04;
    END IF;

WHEN p04 =>     -- midsel_h, iagin_h, sag_h2 active
    IF seqend_h THEN
        tmg_h.s = VCC;
        sag_h2.r = VCC;
        ptsm = p05;
    END IF;

WHEN p05 =>     -- midsel_h, iagin_h, tmg_h active
    IF seqend_h THEN
        tmg_h.r = VCC;
        sag_h1.s = !startup_h;
        sag1sel_h.s = VCC;
        ptsm = p06;
    END IF;

WHEN p06 =>     -- midsel_h, iagin_h active; sag_h1, sag_h2, sag1sel_h, xfrdone_h possibly
active
    IF seqend_h THEN
        iagin_h.r = VCC;
        tmg_h.s = VCC;
        IF sag1sel_h THEN
            sag_h1.r = VCC;
        ELSE
            sag_h2.r = VCC;
        END IF;
        ptsm = p07;
    END IF;

WHEN p07 =>     -- midsel_h, tmg_h active; sag1sel_h, xfrdone_h possibly active
    IF seqend_h THEN
        tmg_h.r = VCC;
        IF sag1sel_h THEN
            sag_h1.s = !startup_h;
        ELSE
            sag_h2.s = !startup_h;
        END IF;
        ptsm = p08;
    END IF;

WHEN p08 =>     -- midsel_h active; sag_h1, sag_h2, sag1sel_h, xfrdone_h possibly active
    IF seqend_h THEN
        tmg_h.s = VCC;
        IF sag1sel_h THEN
            sag_h1.r = VCC;
        ELSE
            sag_h2.r = VCC;
        END IF;
        ptsm = p09;
    END IF;

WHEN p09 =>     -- midsel_h, tmg_h active; sag1sel_h, xfrdone_h possibly active
```

```
            IF seqend_h THEN
                tmg_h.r = VCC;
                IF xfrdone_h THEN
                    sag_h1.s = !startup_h;
                ELSIF sag1sel_h THEN
                    iagin_h.s = VCC;
                    sag_h1.s = !startup_h;
                ELSIF !vcnt243_h THEN
                    iagin_h.s = VCC;
                    sag_h2.s = !startup_h;
                    vcntena_h.d = VCC;
                ELSE
                    sag_h2.s = !startup_h;
                    midsel_h.r = VCC;
                    xfrdone_h.s = VCC;
                END IF;
                ptsm = p10;
            END IF;

        WHEN p10 =>     -- iagin_h active; midsel_h, sag_h1, sag_h2, sag1sel_h, xfrdone_h possibly
active
            IF seqend_h THEN
                tmg_h.s = VCC;
                IF sag1sel_h THEN
                    sag_h1.r = VCC;
                ELSE
                    sag_h2.r = VCC;
                END IF;
                ptsm = p11;
            END IF;

        WHEN p11 =>     -- iagin_h, tmg_h active; midsel_h, sag1sel_h, xfrdone_h possibly active
            startup_h.r = VCC;
            IF seqend_h THEN
                IF !sag1sel_h THEN
                    tmg_h.r = VCC;
                    sag1sel_h.s = VCC;
                    sag_h1.s = VCC;
                    ptsm = p06;
                ELSIF !xfrdone_h THEN
                    tmg_h.r = VCC;
                    sag1sel_h.r = VCC;
                    sag_h2.s = VCC;
                    ptsm = p06;
                ELSE
                    ptsm = p00;
                END IF;
            END IF;

    END CASE;

END;
```

```
%=================================================================
Readout Controller

Kensal Corporation
Phase II Camera Prototype

Author: Ken Crocker
Date:   6 Nov 96
File:   READOUTC.TDF
Rev:    1.00
=================================================================%
TITLE "Phase II Camera Prototype - Readout Controller";

SUBDESIGN readoutc
(
    clk_r               : INPUT;    -- 40 MHz clock
    clr_l               : INPUT;    -- asynchronous reset

    srgon_h             : INPUT;    -- activates srg_h[3..1] on next cycle when idle
    simulating_h        : INPUT;
    allpixels_h         : INPUT;    -- when active, even dark pixels are digitized
    darkline_h          : INPUT;    -- when active, the entire line is a dark line
    go_h                : INPUT;
    sync_h              : INPUT;

    dv_h                : OUTPUT;   -- active when data is valid for conversion

    srg_h[3..1]         : OUTPUT;
    clamp_h[3..1]       : OUTPUT;
    sample_h[3..1]      : OUTPUT;

    busy_h              : OUTPUT;
)

VARIABLE
    dv_h                : SRFF;

    cld1_h[3..1]        : LCELL;
    cld2_h[3..1]        : LCELL;
--  sad1_h[3..1]        : LCELL;
--  sad2_h[3..1]        : LCELL;

    hcnt_h[8..0]        : DFF;
    endline_h           : LCELL;
    enddummy_h          : LCELL;
    startdark_h         : LCELL;

    rosm                : MACHINE OF BITS (
                            hcntclr_h, hcntena_h, busy_h, srg_h[3..1],
                            cl_h[3..1], sa_h[3..1]
                        ) WITH STATES (
--                                  hh
--                                  cc
--                                  nn
--                                  ttb
--                                  ceusss
--                                  lnsrrrcccsss
--                                  rayggglllaaa
--                                  ------------
--                                  hhhhhhhhhhhh
--                                     321321321
                            r00 = B"100000111000",

                            r01 = B"000111111000",
```

```
                        r02 = B"001001100001",
                        r03 = B"001010001010",
                        r04 = B"011100010100",

                        r05 = B"001001100001",
                        r06 = B"001010001010",
                        r07 = B"001000000000"
                    );

BEGIN
    -- Data Valid Calculation
    --      Signal 'dv_h' signals the A/D converter that there is a valid analog signal to
    -- be converted. Referring to the TC217 data book, there are 12 "dummy" pixels for each
    -- of three channels (for a total of 36 pixels) at the start of every row. Signal
    -- 'enddummy_h' is used to set the RS flipflop 'dv_h' and start A/D conversion.
    --      When 'dv_h' is reset depends on the input 'allpixels_h'. If 'allpixels_h' is
    -- active, then 1158 pixels are digitized (386 pixel groups at 3 pixels per group).
    -- The makeup of the line is as follows: 1/2 dark pixel, 1134 active pixels, 1/2 dark
    -- pixel, and 22 fully dark pixels (The TI data book says 23 dark pixels, but that would
    -- require 1159 pixels per line).
    --      If 'allpixels_h' is inactive, then 1140 pixels are digitized (or 380 pixel groups).
    -- The makeup of the line will then be: 1/2 dark pixel, 1134 active pixels, 1/2 dark pixel,
    -- and 4 fully dark pixels.

    dv_h.clk     = clk_r;
    !dv_h.clrn   = !clr_l;

--  clamp_h[]    = cl_h[];

    cld1_h[]     = cl_h[];          -- implemented as an LCELL
    cld2_h[]     = cld1_h[];        -- implemented as an LCELL
    clamp_h[]    = cld2_h[];

    sample_h[]   = sa_h[];

--  sad1_h[]     = sa_h[];          -- implemented as an LCELL
--  sad2_h[]     = sad1_h[];        -- implemented as an LCELL
--  sample_h[]   = sad2_h[];

    -- Horizontal Counter
    hcnt_h[].clk     = clk_r;
    !hcnt_h[].clrn   = !clr_l;
    IF hcntclr_h THEN
        hcnt_h[].d = 0;
    ELSIF hcntena_h THEN
        hcnt_h[].d = hcnt_h[].q + 1;
    ELSE
        hcnt_h[].d = hcnt_h[].q;
    END IF;

    IF simulating_h THEN
        endline_h = (hcnt_h[].q == 25);     -- LCELL. Imaginary sensor with 26 pixel groups per line (26
- 12 = 14 x 3 = 42 pixels)
    ELSE
        endline_h = (hcnt_h[].q == 397);    -- LCELL. Real TC217 sensor with 398 pixel groups per line
(398 - 12 = 386 x 3 = 1158 pixels)
    END IF;
    enddummy_h  = (hcnt_h[].q == 12);       -- LCELL. There are 12 dummy pixels at the start
                                            -- of each line. We use 12 instead of 11
                                            -- because the 'enddummy_h' LCELL is used
                                            -- AFTER incrementing to the 13th pixel.

    startdark_h = (hcnt_h[].q == 392);      -- LCELL. 12 dummy groups + 380 digitized groups = 392.
```

```
rosm.clk     = clk_r;
rosm.reset   = !clr_l;

%===========================================================================
Readout State Machine
===========================================================================%
CASE (rosm) IS
    -- Idle (integration)
    WHEN r00 =>             -- hcntclr_h, cl_h[] (sync_h) active
        IF srgon_h THEN rosm = r01;
        ELSIF go_h & sync_h THEN rosm = r02;
        END IF;

    -- Activate srg's during vertical shift
    WHEN r01 =>             -- srg_h[1..3], cl_h[] active
        IF !srgon_h THEN rosm = r00;
        END IF;

    -- Start of horizontal readout
    WHEN r02 =>             -- busy_h, srg_h1, cl_h3, sa_h1 active
        rosm = r03;
    WHEN r03 =>             -- busy_h, srg_h2, cl_h1, sa_h2 (adclk_r1) active
        rosm = r04;
    WHEN r04 =>             -- hcnten, busy_h, srg_h3, cl_h2, sa_h3 (sync_h) active
        dv_h.s = enddummy_h & (allpixels_h # !darkline_h);
        dv_h.r = startdark_h & !allpixels_h;
        IF endline_h THEN rosm = r05;
        ELSE rosm = r02;
        END IF;

    WHEN r05 =>             -- busy_h, srg_h1, cl_h3, sa_h1 active
        rosm = r06;
    WHEN r06 =>             -- busy_h, srg_h2, cl_h1, sa_h2 active
        rosm = r07;
    WHEN r07 =>             -- busy_h (sync_h) active
        dv_h.r = VCC;
        rosm = r00;
END CASE;

END;
```

KENSAL CORPORATION TRADE SECRETS

Frame Capture PCB Register Definitions

Revision D: October 26, 1998

Frame Capture Board - Base Address Register Assignments

BAR	Description
1	Altera Write Controller (32 bytes)
0	Altera PCI & Read/Write Controller (64 bytes)

Altera PCI & Read/Write Controller Assignments

Addr	Name		Special Conditions
0x24	Rd Line Counter/Register	[10..00]	
0x20	Rd/Wr Address Counter/Register	[20..00]	
0x1C	Rd/Wr Address Increment	[12..00]	
0x18	Rd/Wr Pixel Counter/Register	[12..00]	Read only.
0x14	Rd/Wr Pixel Register	[12..00]	
0x10	PCI Line Counter/Register	[10..00]	Write only in current version.
0x0C	PCI Address Register	[31..02]	LS 2-bits are always '0'
0x08	PCI Address Increment	[13..02]	LS 2-bits are always '0'
0x04	PCI Transfer Counter Register	[14..02]	LS 2-bits are always '0'
0x00	Control/Status Register		

Rd Line Counter/Register

This counter/register tells the Rd/Wr controller how many lines of video to read from the frames capture board's DRAM. It is not used during diagnostic writes to DRAM because the number of video lines is controlled by the PCI side (specifically by the PCI Line Counter/Register). Writing a non-zero value to this register starts transfers from video DRAM into the FIFO within the S5933Q.

Rd/Wr Address Counter/Register

This counter contains the address where a read or a write from/to frame capture DRAM will occur next. It is automatically incremented after every read/write. When writing to this register, the LS bit should always be '0'.

Rd/Wr Address Increment

This register contains the address increment to use at the end of each video line to add to the Rd/Wr Address Counter/Register in order to point to the first pixel of the next line.

Rd/Wr Pixel Counter/Register

This read-only counter gets loaded with contents of the Rd/Wr Pixel Register at the start of every line. It decrements by one for every 32-bits transferred. When an attempt is made to decrement it past zero, an end-of-line condition is indicated.

Rd/Wr Pixel Register

This register contains the number of 32-bit transfers per line of video minus 1. For example, suppose there were 1024 16-bit pixels per line. This register would be loaded with 511 since there are 2 16-bit pixels packed in one 32-bit longword. One 32-bit longword can also hold either one 24-bit RGB pixel or four 8-bit data mode pixels.

PCI Line Counter/Register

This counter indicates the number of lines to transfer to either main Mac DRAM or video DRAM on the video adaptor using master mode transfers. Master mode transfers imply that the Frame Capture Board is the DMA master, handling all of the transfer details. Alternatively, the Frame Capture Board may be accessed as a PCI target. In this case, the Rd/Wr registers described above are used, but the PCI registers are not. Master mode transfers are initiated whenever the contents of this register changes from 0 to a non-zero value. Because of this, all other registers must be set up before writing a non-zero value to this register. If an error occurs (eg. master/target abort, see below), the register will contain the number of lines left to transfer. To re-initiate transfers, one must write a 0 to the register, then a non-zero value. Note that whenever the contents of this register decrements from 1 to 0, a 'dmadone_h' interrupt is generated.

PCI Address Register

This register contains the starting address within the Mac's main DRAM or the Video Adaptor's DRAM where transfers are to take place. When master mode transfers are enabled, the contents of this register are written into the MWAR or MRAR of the S5933 PCI Controller, depending on whether write or read transfers are requested. At the end of every line, the following steps occur: a) the MWAR or MRAR register is read into the PCI Address Register, b) the PCI Address Increment is added to the value contained in this register, and, c) the new value of the PCI Address Register is written back into the MWAR or MRAR register in preparation for the transfer of the next line of data. Therefore, at the end of the transfer, the PCI Address Register would contain the address of the beginning of the last line of video.

PCI Address Increment

This register is added to the S5933 PCI Controller's MWAR or MRAR register at the end of every line transferred to point to the start of the next line of video.

PCI Transfer Count Register

This register contains the number of bytes per video line to be transferred. For example, if a line of video contains 1024 16-bit pixels, then this register would be loaded with 2048. At the start of each video line transfer, the S5933 PCI Controller's MWTC or MRTC is loaded with the contents of this register.

PCI & Read/Write Controller Control/Status Register Bit Definitions

Bit	Name	Write	Read
31			
30			
29			
28			
27			
26			
25			
24			
23			
22			
21			
20			
19			
18			
17			
16			
15			
14			
13			
12			
11			
10	dmadone_h	0 -> clear bit, 1 -> NOP	DMA Done Interrupt
9	mtabort_h	0 -> clear bit, 1 -> NOP	Master/Target Abort Interrupt
8	bistint_h	0 -> clear bit, 1 -> NOP	BIST Interrupt
7	irqerr_h	0 -> clear bit, 1 -> NOP	IRQ Error Interrupt
6	rwinten_h	R/W Interrupt Enable	R/W Interrupt Enable
5	rwint_h	<read only>	R/W Interrupt
4	write_h	Write Mode	Write Mode
3	depth_h[1]	Depth Mode Bit 1	Depth Mode Bit 1
2	depth_h[0]	Depth Mode Bit 0	Depth Mode Bit 0
1	color_h[1]	Color Mode Bit 1	Color Mode Bit 1
0	color_h[0]	Color Mode Bit 0	Color Mode Bit 0

color_h[1..0] Color Mode

These bits select the color mode for reads and writes between the PCI bus and the video DRAM on the Frame Capture PCB. There is some interaction between color_h[] and depth_h[], in that some combinations are disallowed or only allowed for data transfers in certain directions only (see Special Conditions below). The value written may be read back by reading the control register.

color_h[1..0]	Description	Special Conditions
11	RGB	Not allowed for Data-8 transfers
10	Monochrome Blue	
01	Monochrome Green	
00	Monochrome Red	

depth_h[1..0] Depth Mode

These bits select the pixel depth for reads and writes between the PCI bus and the video DRAM on the Frame Capture PCB. The separate document, entitled "Frame Capture Board Video/Data Transfers" graphically illustrates how color_h[] and depth_h[] affect transfers. To summarize, the Frame Capture Board supports 16 and 32-bit video pixels (Video-16 and Video-32). The Data-8 mode is a diagnostic mode used for testing video DRAM. The value written may be read back by reading the control register.

depth_h[1..0]	Description	Special Conditions
11	Reserved	
10	Data-8	Only monochrome transfers supported.
01	Video-32	For PCI->FCB transfers, only RGB is sup.
00	Video-16	PCI->FCB transfers not supported

write_h Write Mode

This bit selects the direction of master mode transfers to/from the Frame Capture board. Write transfers use the S5933 Bus Master Write Address Register and Bus Master Write Transfer Count, along with the Add-on Master Write Enable (AMWEN) pin. Note that master mode write operations transfer data in the same direction as slave mode reads, that is, from the Frame Capture PCB to the PCI bus. Conversely, when write_h is inactive, the S5933 Bus Master Read Address Register and Bus Master Read Transfer Count, along with the Add-on Master Read Enable (AMREN) pin are used. This pin is sampled at the start of transfers only. The value written may be read back by reading the control register.

rwint_h R/W Interrupt

This status bit goes high whenever one or more of 'irqerr_h', 'bistint_h', mtabort_h' or 'dmadone_h' bits are set, indicating an interruptable condition. To generate an actual PCI hardware interrupt, the following conditions must also be met: 1) the 'rwinten_h' bit must be set, and, 2) the appropriate bits in the Interrupt Control/Status Register (INTCSR) must be set to enable interrupts on incoming mailbox #4, byte #0. See the section on Interrupt Generation below for details. This bit is cleared when all of the interrupt sources mentioned above have been cleared.

rwinten_h R/W Interrupt Enable

When this bit is active, 'rwint_h' is propagated out of the Read/Write Controller to the S5933, potentially triggering INTA# on the PCI bus. See the section on Interrupt Generation below for details. The value written may be read back by reading the control register.

irqerr_h IRQ Error Interrupt

This status bit goes active when the addon circuitry attempts to clear it's addon interrupt line IRQ#, and the attempt fails (ie. IRQ# stays asserted). It should never happen but the bit is included since, technically, it *could* happen. This bit is cleared by writing a '0' into it. Writing a '1' has no effect.

bistint_h BIST Interrupt

When this status bit is active, the addon circuitry has received an interrupt from the S5933 indicating that the host wants to do a "Built In Self Test" function. This feature is currently unsupported by the Frame Capture Board. If you don't issue a PCI BIST command, this bit should never become active. This bit is cleared by writing a '0' into it. Writing a '1' has no effect.

mtabort_h Master/Target Abort Interrupt

When this status bit is active, the addon circuitry has received an interrupt from the S5933 indicating that an error occurred during the DMA. Transfers are automatically halted. Under normal circumstances (ie. we've gotten DMA to work properly), this bit should never get set. This bit is cleared by writing a '0' into it. Writing a '1' has no effect.

dmadone_h DMA Done Interrupt

This bit goes active whenever the PCI Line Counter/Register transitions from '1' to '0' during DMA transfers, indicating that the PCI Controller has just finished master mode transfers and entered it's idle state. Note that the end of DMA transfers sets this bit, not the fact that the register contains zero. This bit is cleared by writing a '0' into it. Writing a '1' has no effect.

Altera Write Controller Assignments

Addr	Name	Special Conditions
0x14	Write Address Counter/Register	[19..00]
0x10	Write Address Increment	[12..00]
0x0C	Write Pixel Counter/Register	[12..00]. Read only.
0x08	Write Pixel Register	[12..00]
0x04	Status Register	Mostly read only. Some bits cleared by write.
0x00	Control Register	

Write Address Counter/Register

This counter contains the address where a write to frame capture DRAM will occur next. The memory is organized as 3 (R, G, & B) x 16-bits (LS & MS pixels) x 1 Meg. Subsequently, from the Write Controller's point of view there are two 24-bit pixels per memory address. The counter/register automatically increments when the "last" MS pixel is written (varies, depending on write mode).

Write Address Increment

This register contains the address increment to use at the end of each video line to add to the Write Address Counter/Register in order to point to the first pixel of the next line. For interlaced video, this register should be loaded with the number of pixels per line / 2.

Write Pixel Counter/Register

This read-only counter gets loaded with contents of the Write Pixel Register at the start of every line. It decrements by one whenever the Write Address Counter/Register is incremented. When an attempt is made to decrement it past zero, an end-of-line condition is indicated.

Write Pixel Register

This register contains the number of pixels per line / 2 minus 1. For example, suppose there were 1024 pixels per line. This register would be loaded with 511.

Write Controller Control Register Bit Definitions

Bit	Name	Write	Read
31			
30			
29			
28			
27			
26	spare_h[1]	Spare Bit 1 (yel LED)	Spare Bit 1
25	spare_h[0]	Spare Bit 0 (red LED)	Spare Bit 0
24	tsc_h	Tx Select Cmd	Tx Select Cmd
23	td_h[7]	Tx Data Bit 7	Tx Data Bit 7
22	td_h[6]	Tx Data Bit 6	Tx Data Bit 6
21	td_h[5]	Tx Data Bit 5	Tx Data Bit 5
20	td_h[4]	Tx Data Bit 4	Tx Data Bit 4
19	td_h[3]	Tx Data Bit 3	Tx Data Bit 3
18	td_h[2]	Tx Data Bit 2	Tx Data Bit 2
17	td_h[1]	Tx Data Bit 1	Tx Data Bit 1
16	td_h[0]	Tx Data Bit 0	Tx Data Bit 0
15	<reserved>	<reserved>	<reserved>
14	twr_h	Tx Write	0
13	rfrd_h	Rx FIFO Read	Rx FIFO Read (clrd after FIFO rd)
12	writebank1_h	Write DRAM Bank 1	Write DRAM Bank 1
11	wrinten_h	Write Interrupt Enable	Write Interrupt Enable
10	xfren_h	Transfer Enable	Transfer Enable (clrd when xfr done)
9	mode_h[1]	Mode Bit 1	Mode Bit 1
8	mode_h[0]	Mode Bit 0	Mode Bit 0
7	rrf_h	Rx Reframe Enable	Rx Reframe Enable
6	rselb_h	Rx Select "B" Input	Rx Select "B" Input
5	tsvs_h	Tx Send Violation Symbol	Tx Send Violation Symbol
4	tenn_h	Tx Enable Next Word	Tx Enable Next Word
3	rbisten_h	Rx BIST Enable	Rx BIST Enable
2	tbisten_h	Tx BIST Enable	Tx BIST Enable
1	rfrst_h	Rx FIFO Reset	0
0	enrfwen_h	Enable Rx FIFO Writes	Enable Rx FIFO Writes

enrfwen_h Enable Rx FIFO Writes

When active, this bit enables circuitry that takes received data words from the Hotlink receiver and writes them to the receive FIFO. The value written may be read back by reading the control register.

rfrst_h Rx FIFO Reset

When a '1' is written into this bit, the receive FIFO is reset to an empty state. This bit is automatically cleared, so a value of '0' will always be read.

tbisten_h Tx BIST Enable

An inverted form of this signal is tied directly to the BISTEN_L bit of the Hotlink Transmitter. The value written may be read back by reading the control register.

rbisten_h Rx BIST Enable

An inverted form of this signal is tied directly to the BISTEN_L bit of the Hotlink Receiver. The value written may be read back by reading the control register.

tenn_h   Tx Enable Next Word
An inverted form of this signal is tied directly to the ENN_L bit of the Hotlink Transmitter. It is used during BIST mode to enable the start of the cyclic test pattern. The value written may be read back by reading the control register.

tsvs_h   Tx Send Violation Symbol
When a '1' is written to this bit, the next data word write to the Hotlink Transmitter will have the SVS_H bit set.

rselb_h   Rx Select "B" Input
One of the Hotlink transmitter outputs is hardwired to one of the Hotlink receiver inputs as a test feature. When this bit is active, the receiver input connected to the loopback is selected, enabling a thorough test of almost all of the electronics without having an attached camera head or an external loopback cable. The value written may be read back by reading the control register.

rrf_h   Rx Reframe Enable
This bit is directly connected to the RF_H input of the Hotlink receiver. This bit should normally have a value of '1'. See the Hotlink data sheet for details on the operation of this bit. The value written may be read back by reading the control register.

mode_h[1..0]   Mode Bits
The mode bits control how data is stored in video DRAM. For the TC217 and RL4000P, data are received from one color (R, G, or B) at a time. For the Kodak tri-color linescan array, data are received for all three primary colors with the same exposure. The data are received in the byte stream as R, G, B, F, R, G, B, F, etc. The 'F' byte represents a filler byte of 0x00. The value written may be read back by reading the control register.

00 --> Monochrome Red (received bytes are all of red color)
01 --> Monochrome Green (received bytes are all of green color)
10 --> Monochrome Blue (received bytes are all of blue color)
11 --> RGB Mode (received bytes are in R, G, B, F, R, G, B, F... sequence, where 'F' is a filler byte of 0x00)

xfren_h   Transfer Enable
When active, this bit enables transfers from the receive FIFO into the video DRAM according to the currently selected mode. The Write Pixel Register, Write Address Increment, and Write Address Counter should have been set up previously. Transfers will continue until a Hotlink control symbol is encountered, at which time this bit will be cleared and an interrupt generated. The actual control byte will be in rfd_h[] (see below). There is no need to activate the 'rfrd_h' bit (see below) to get it. The value written may be read back by reading the control register. An ongoing transfer may be aborted by writing a '0' to this bit. The FIFO will continue to fill, an interrupt will be generated (if enabled), and the 'oddbyte_h' status bit will be undefined.

wrinten_h   Write Interrupt Enable
When this bit is active, 'wrint_h' is propagated out of the Write Controller to the S5933, potentially triggering INTA# on the PCI bus. See the section on Interrupt Generation below for details. The value written may be read back by reading the control register.

writebank1_h   Write DRAM Bank #1
There are two DRAM banks, signified #0 and #1. When this bit is active, the Write Controller is "hooked up" to bank #1. The Read/Write Controller is therefore attached to bank #0. When this bit is inactive, the roles are reversed, with the Write Controller going to bank #0.

rfrd_h Rx FIFO Read

After a video DRAM transfer is started, transfers will continue until a control byte is read from the receive FIFO. It may be necessary to continue to manually read control/data bytes from the receive FIFO. When a '1' is written to this bit, the next control/data byte is read from the receive FIFO. It can be accessed by the rfd_h[] bits in the status register. If the transmit FIFO is empty when a '1' is written to this bit, the FIFO read will occur as soon as data is written to the FIFO. This bit is cleared after the data are read from the FIFO.

twr_h Tx Write

When a '1' is written to this bit, the value of the td_h[] bits is written into the transmit FIFO. Since the write happens instantaneously, this bit will always return a '0' if read.

td_h[7..0] Tx Data

These 8 bits are written to the transmit FIFO when a '1' is written to 'twr_h'. The value written may be read back by reading the control register.

tsc_h Tx Select Cmd

When this bit is active, bits td_h[] are interpreted as a command -- when inactive, the bits are interpreted as data. The value written may be read back by reading the control register.

spare_h[1..0] Spare Bits

These bits go out to test points and LED's on the PCB. Writing a '1' to a bit will cause a low level to be outputted to the corresponding test point (and the corresponding LED to illuminate) - a '0' will cause a high level. The value written may be read back.

Write Controller Status Register Bit Definitions

Bit	Name	Write	Read
31			
30			
29			
28			
27			
26			
25			
24	rfsc_h	<read only>	Rx FIFO Select Command
23	rfd_h[7]	<read only>	Rx FIFO Data Bit 7
22	rfd_h[6]	<read only>	Rx FIFO Data Bit 6
21	rfd_h[5]	<read only>	Rx FIFO Data Bit 5
20	rfd_h[4]	<read only>	Rx FIFO Data Bit 4
19	rfd_h[3]	<read only>	Rx FIFO Data Bit 3
18	rfd_h[2]	<read only>	Rx FIFO Data Bit 2
17	rfd_h[1]	<read only>	Rx FIFO Data Bit 1
16	rfd_h[0]	<read only>	Rx FIFO Data Bit 0
15			
14			
13			
12			
11			
10			
9			
8			
7	wrint_h	0 -> clear bit, 1 -> NOP	Write Interrupt
6	oddbyte_h	0 -> clear bit, 1 -> NOP	Odd Number of Bytes Transferred
5	trplat_h	0 -> clear bit, 1 -> NOP	Tx Hotlink RP Pulse Detected
4	rcd_h	<read only>	Rx Carier Detect
3	rrvs_h	0 -> clear bit, 1 -> NOP	Rx Received Violation Symbol
2	rfov_h	0 -> clear bit, 1 -> NOP	Rx FIFO Overflowed
1	rfor_h	<read only>	Rx FIFO Output Ready
0	rfir_h	<read only>	Rx FIFO Input Ready

rfir_h Rx FIFO Input Ready

When active, this bit indicates that the receive FIFO is not full.

rfor_h Rx FIFO Output Ready

When active, this bit indicates that the receive FIFO is not empty.

rfov_h Rx FIFO Overflowed

When active, this bit indicates that a write was attempted to the receive FIFO when it was full. This bit is cleared by writing a '0' into it. Writing a '1' has no effect.

rrvs_h Rx Received Violation Symbol

When active, this bit indicates that a Hotlink word was received with the RVS_H bit set. This bit is cleared by writing a '0' into it. Writing a '1' has no effect.

rcd_h Rx Carrier Detect

When active, this bit indicates that there is activity on the primary receive channel.

trplat_h Tx Hotlink RP Pulse Detected
When the Tx Hotlink is in BIST mode, one of its signal pins (RP) will pulse in between executions of its self test transmission loop. This pin will go active when the pulse is detected. In normal mode, the RP signal will be a 33 MHz clock, so this pin will be always be high. This bit is cleared by writing a '0' into it. Writing a '1' has no effect.

oddbyte_h Odd Number of Bytes Transferred
This bit is set at the end of a transfer when an odd number of bytes was received by the Hotlink receiver. This always signifies an error condition, since an even number of bytes should always be sent. This bit is cleared by writing a '0' into it. Writing a '1' has no effect.

wrint_h Write Interrupt Active
This bit is set whenever the falling edge of 'xfren_h' is detected. To generate an actual PCI interrupt, the following conditions must also be met: 1) the 'wrinten_h' bit must be set, and, 2) the appropriate bits in the Interrupt Control/Status Register (INTCSR) must be set to enable interrupts on incoming mailbox #4, byte #0. See the section on Interrupt Generation below for details. This bit is cleared by writing a '0' into it. Writing a '1' has no effect.

rfd_h[7..0] Rx FIFO Data
These 7-bits contained the received control/data byte from receive FIFO. At the end of a transfer (see 'xfren_h' above) the control byte that caused the end of the transfer will be in 'rfd_h[]'. Bit rfsc_h will be set, indicating that 'rfd_h[7..0]' contain a control byte. Subsequent control/data bytes may be plucked from the pipeline by writing a '1' to the control register bit 'rfrd_h'. Note that once a byte is read, it cannot be "unread", so it is important that the protocol "know" when it must read a byte manually.

rfsc_h Rx FIFO Select Command
When active, this bit signifies that the bits in rfd_h[] specify one of the eleven or so control codes instead of a data byte.

Interrupt Generation

PCI interrupts from the Frame Capture PCB may be generated from either the Write Controller or the Read/Write Controller. In either case, the resulting PCI interrupt used is INTA#. The Write Controller interrupt 'wrint_h' is triggered by the reception of a command symbol on the Hotlink interface. These symbols are placed at the start and end of every "packet" of data. The "interrupt" is always generated, ie. the 'wrint_h' bit is always set when a symbol is received. Whether or not it propagates to become an actual hardware PCI interrupt is dependent on several factors. First, each interrupt has it's own enable bit. For example, setting the 'wrinten_h' bit allows the 'wrint_h' interrupt to propagate at least as far as the S5933 PCI Controller. If the S5933 is configured appropriately, the interrupt will further propagate onto the PCI bus.

The Read/Write Controller interrupt 'rwint_h' activates whenever one of the following bits goes active: 'irqerr_h', 'bistint_h', mtabort_h' or 'dmadone_h'. Like the Write Controller interrupt, it also has its own enable bit, 'rwinten_h'.

The S5933 has many ways to generate an interrupt. Since we are using add-on initiated bus mastering, we will use the mailbox technique. Some details may be found in the Spring 1996 "S5933 PCI Controller Data Book", section 9.1.2, beginning on page 9-4. The S5933 may be programmed to trigger an interrupt when a particular incoming mailbox is written to from the add-on side. In our case, we will use mailbox #4, byte #0. Bit 0 (ie. the LSB in the 32-bit mailbox) will be set if a Write Controller interrupt is active - bit 1 (ie.

0x00000002) for a Read/Write Controller interrupt. To prevent race conditions, the Frame Capture board will continually write to mailbox #4 whenever an interrupt is active. Because of this, to clear the interrupt(s) you must first clear the appropriate interrupt bits ('wrint_h' or 'rwint_h'), then read mailbox #4 to clear the PCI interrupt. Ordering is important here - if you don't clear the source of the interrupt, the mailbox will generate interrupts continuously.

Interrupt software should also also include code to check for errors that may have occurred during DMA transfers, such as a Target or Master Abort (see the databook, page 4-11). The Interrupt Control/Status Register (INTCSR) can also be used to determine that the PCI interrupt has been deasserted (bit 23, "Interrupt Asserted"). The PCI Status Register (PCISTS, see page 3-8) should also be checked for errors. If errors are encountered, appropriate error dialogs should be displayed and the errors cleared.

```
%==========================================================================
PCI.TDF - PCI I/F Controller

Revision History:
1.00     K.W. Crocker    23 Jul 96
    Initial writing.
1.01     K.W. Crocker    29 Nov 96
    Added rwdoe_h output.
1.02     K.W. Crocker    6 Sep 97
    Added ability to read 'pciLineCnt'. Changed interrupt scheme.
==========================================================================%
TITLE "PCI Interface Controller";

INCLUDE "rdwrctlr.inc";

-- Values for ptnum_h[1..0]
CONSTANT rwCtrl          = B"00";     -- PCI & Read/Write Controller (ie. this chip)
CONSTANT wrCtrl          = B"01";     -- Write Controller (ie. not this chip)

-- Longword offset within passthru region
CONSTANT pciCtrlStat     = B"0000";  -- 0x00 >> 2 = 0x0
CONSTANT pciTCReg        = B"0001";  -- 0x04 >> 2 = 0x1
CONSTANT pciAdrInc       = B"0010";  -- 0x08 >> 2 = 0x2
CONSTANT pciAdrReg       = B"0011";  -- 0x0C >> 2 = 0x3
CONSTANT pciLineCnt      = B"0100";  -- 0x10 >> 2 = 0x4
CONSTANT rwPixReg        = B"0101";  -- 0x14 >> 2 = 0x5
CONSTANT rwPixCnt        = B"0110";  -- 0x18 >> 2 = 0x6 (Read only)
CONSTANT rwAdrInc        = B"0111";  -- 0x1C >> 2 = 0x7
CONSTANT rwAdrCnt        = B"1000";  -- 0x20 >> 2 = 0x8
CONSTANT rdLineCnt       = B"1001";  -- 0x24 >> 2 = 0x9

-- adr_h[6..2] Constants
CONSTANT AOMB4           = B"00111"; -- 0x1C >> 2 = 0x07
CONSTANT MWAR            = B"01001"; -- 0x24 >> 2 = 0x09
CONSTANT APTD            = B"01011"; -- 0x2C >> 2 = 0x0B
CONSTANT MRAR            = B"01100"; -- 0x30 >> 2 = 0x0C
CONSTANT AINT            = B"01110"; -- 0x38 >> 2 = 0x0E
CONSTANT AGCSTS          = B"01111"; -- 0x3C >> 2 = 0x0F
CONSTANT MWTC            = B"10110"; -- 0x58 >> 2 = 0x16
CONSTANT MRTC            = B"10111"; -- 0x5C >> 2 = 0x17

SUBDESIGN pci (
    -- AMCC S5933 System Signals
    bpclk_r              : INPUT;    -- buffered PCI clock
    sysrst_l             : INPUT;    -- system reset
    irq_l                : INPUT;    -- interrupt request from S5933

    -- AMCC S5933 Passthru Interface Signals
    ptatn_l              : INPUT;    -- pass thru cycle input
    ptburst_l            : INPUT;    -- burst access input
    ptnum_h[1..0]        : INPUT;    -- base address register number
    ptbe_l[3..0]         : INPUT;    -- requested byte enables
    ptwr_h               : INPUT;    -- write (ptrd_l) input
    ptadr_l              : OUTPUT;   -- address reg select (slow slew rate)
    ptrdy_l              : OUTPUT;   -- ready output (slow slew rate)

    -- AMCC S5933 Add-On Bus Interface Signals
    dq_h[31..00]         : BIDIR;    -- data bus (slow slew rate)
    adr_h[6..2]          : OUTPUT;   -- register address (slow slew rate)
    select_l             : OUTPUT;   -- cycle start (slow slew rate)
    rd_l                 : OUTPUT;   -- read strobe (slow slew rate)
    wr_l                 : OUTPUT;   -- write strobe (slow slew rate)
    pcibusgrnt_l         : BIDIR;    -- addon bus grant for pci controller
    rwbusgrnt_l          : BIDIR;    -- addon bus grant for read/write controller
```

```
    -- AMCC S5933 Addon DMA Interface Signals
    amwen_h             : OUTPUT;   -- addon bus mastering write enable
    amren_h             : OUTPUT;   -- addon bus mastering read enable

    -- AMCC S5933 Mailbox Signals
    emb_h[7..0]         : OUTPUT;   -- connected to ea[7..0] (slow slew rate)
    embclk_r            : OUTPUT;   -- connected to ea[8] (slow slew rate)

    -- AMCC S5933 FIFO Signals
    wrfifo_l            : OUTPUT;   -- write fifo strobe
    rdfifo_l            : OUTPUT;   -- read fifo strobe
    wrfull_h            : INPUT;    -- write fifo full input
    rdempty_h           : INPUT;    -- read fifo empty input

    -- Interface with Super Mux
    rwmuxsel_h[4..0]    : OUTPUT;
    rwa_h[9..0]         : OUTPUT;
    rwdoe_h             : OUTPUT;
    rwras_h             : OUTPUT;   -- slow slew rate
    rwcas_h             : OUTPUT;   -- slow slew rate
    rwallras_h          : OUTPUT;   -- slow slew rate
    rwallcas_h          : OUTPUT;   -- slow slew rate
    rwwe_h              : OUTPUT;   -- slow slew rate
    rwor_h              : INPUT;    -- Super Mux FIFO's output ready
    rwren_h             : OUTPUT;   -- Super Mux FIFO's read enable

    -- Write Controller Signals
    wrint_h             : INPUT;    -- write controller interrrupt
)

VARIABLE
    -- Global and System Signals
    clk_r                   : NODE;
    clr_l                   : NODE;

    -- Read/Write Controller
    dqbusreq_h              : LCELL;
    rdwr                    : rdwrctlr;
    wrsel_h                 : NODE;     -- chip decode

    -- S5933 Pass Thru Signals
    ptatn_h                 : LCELL;
    ptburst_h               : LCELL;
    pcisel_h                : LCELL;
    ptrdytri_l              : TRI;
    ptadrtri_l              : TRI;

    -- S5933 Addon Bus Signals
    dqbuf_h[31..00]         : LCELL;
    dqtri_h[31..00]         : TRI;
    dqoe_l                  : LCELL;
    adrtri_h[6..2]          : TRI;
    selecttri_l             : TRI;
    rdtri_l                 : TRI;
    wrtri_l                 : TRI;
    busgrnttri_l            : TRI;

    -- Passthru Address Register
    ptaddr_h[3..0]          : DFF;      -- offset within passthru region
    ptadrinc_h              : NODE;

    -- Control and Status Register Signals
    ctrlwr_h                : LCELL;    -- control register write enable
```

```
-- Control/Status Register
color_h[1..0]            : DFFE;
depth_h[1..0]            : DFFE;
write_h                  : DFFE;
rwint_h                  : LCELL;
rwinten_h                : DFFE;
irqerr_h                 : SRFF;
bistint_h                : SRFF;
mtabort_h                : SRFF;
dmadone_h                : SRFF;

-- PCI Transfer Count Register
pcitcreg_h[14..02]       : DFF;      -- PCI longword transfers/video line
pcitcregwr_h             : LCELL;

-- PCI Address Register
pciadrreg_h[31..02]      : DFF;      -- PCI longword addresses
pciadrregwr_h            : LCELL;

-- PCI Address Increment
pciadrinc_h[13..02]      : DFF;      -- PCI address increment to start of next line
pciadrincwr_h            : LCELL;

-- Summation Register
sum_h[31..02]            : LCELL;    -- Sum of pci adr cntr and pci adr incr
carry1_h                 : LCELL;
carry2_h                 : LCELL;

-- PCI Line Counter
pcilinecnt_h[10..00]     : DFF;      -- Number of lines to transfer
pcilinecntwr_h           : LCELL;
pcilinecnttc_h           : NODE;
done_h                   : NODE;

-- Pass-Through State Machine
ptSM                     : MACHINE OF BITS (
                             busgrnt_h, select_h, rd_h, wr_h,
                             ptadr_h, ptrdy_h, dqoe_h, tsoe_l,
                             addonintack_h, addonwrack_h, addonrdack_h
                           ) WITH STATES (
                             pt00     = B"00000001000",
                             pt01     = B"11001000000",
                             pt02     = B"11001000000",
                             pt03     = B"11100000000",
                             pt04     = B"11100100000",
                             pt05     = B"11010010000",
                             pt06     = B"11010110000",
                             pt07     = B"10000010100",
                             pt08     = B"11010010100",
                             pt09     = B"10000010010",
                             pt10     = B"11010010010",
                             pt11     = B"11100000001",
                             pt12     = B"00000000000"
                           );

pcilinecnttcdly_h        : DFF;
go_h                     : NODE;
addonwr_h                : SRFF;
addonrd_h                : SRFF;
add_h                    : LCELL;
checkirq_h               : NODE;
checkints_h              : NODE;
loadpciadrreg_h          : NODE;
```

```
    irqcnt_h[1..0]          : DFF;
    irqcnttc_h              : LCELL;
    xfren_h                 : NODE;
    rwSM                    : MACHINE OF BITS (
                                 AGCSTS_h, AINT_h, adrreg_h, tcreg_h,
                                 decrline_h, doaddonwr_h
                              ) WITH STATES (
                                 rw00     = B"000000",
                                 rw01     = B"000000",
                                 rw02     = B"100001",
                                 rw03     = B"100000",
                                 rw04     = B"010001",
                                 rw05     = B"010000",
                                 rw06     = B"001001",
                                 rw07     = B"001000",
                                 rw08     = B"000101",
                                 rw09     = B"000100",
                                 rw10     = B"000000",
                                 rw11     = B"000000",
                                 rw12     = B"000000",
                                 rw13     = B"010000",
                                 rw14     = B"010000",
                                 rw15     = B"000010",
                                 rw16     = B"000000",
                                 rw17     = B"001000",
                                 rw18     = B"001000",
                                 rw19     = B"000000",
                                 rw20     = B"000000"
                              );

    mbff_h[1..0]            : DFFE;
    mbcnt_h[2..0]           : DFFE;
    mbcnttc_h               : NODE;
    mbSM                    : MACHINE OF BITS (
                                 mbena_h, addonint_h, AOMB4_h, doaddonint_h
                              ) WITH STATES (
                                 mb00     = B"1000",
                                 mb01     = B"0000",
                                 mb02     = B"0100",
                                 mb03     = B"0111",
                                 mb04     = B"0110",
                                 mb05     = B"0000",
                                 mb06     = B"1000"
                              );

BEGIN
    -- Global Inputs
    clk_r                = GLOBAL (bpclk_r);
    clr_l                = GLOBAL (sysrst_l);

    %=======================================================================
    Read/Write Controller
    =======================================================================%
    dqbusreq_h = !ptatn_l # addonwr_h # addonrd_h # addonint_h;      -- LCELL

    rdwr.(bpclk_r, sysrst_l, dq_h[20..00], wrfull_h, rdempty_h) =
        (bpclk_r, sysrst_l, dq_h[20..00], wrfull_h, rdempty_h);
    rdwr.(dqbusreq_h, ptaddr_h[3..0], wrsel_h, rwor_h, depth_h[1..0], color_h[1..0]) =
        (dqbusreq_h, ptaddr_h[3..0], wrsel_h, rwor_h, depth_h[1..0], color_h[1..0]);
    rdwr.rfshforce_h = GND;

    (dqbuf_h[31..00], rwbusgrnt_l, wrfifo_l, rdfifo_l, rwmuxsel_h[], rwa_h[9..0]) =
        rdwr.(dqout_h[31..00], busgrnt_l, wrfifo_l, rdfifo_l, rwmuxsel_h[], rwa_h[9..0]);
    (rwdoe_h, rwras_h, rwcas_h, rwallras_h, rwallcas_h, rwwe_h, rwren_h) =
```

```
    rdwr.(rwdoe_h, rwras_h, rwcas_h, rwallras_h, rwallcas_h, rwwe_h, rwren_h);

%===============================================================
Pass Thru Transactions
===============================================================%

-- Passthru Address Register
ptaddr_h[].clk      = clk_r;
!ptaddr_h[].clrn    = !clr_l;
IF pt02 THEN
    ptaddr_h[].d = dq_h[05..02];
ELSIF ptadrinc_h THEN
    ptaddr_h[].d = ptaddr_h[].q + 1;
ELSE
    ptaddr_h[].d = ptaddr_h[].q;
END IF;
ptadrinc_h = ptrdy_h & ptatn_h;

-- Input Bits
ptatn_h     = !ptatn_l;                   -- (LCELL) compensate for clk buf and 0 hld
ptburst_h   = !ptburst_l;                 -- (LCELL) compensate for clk buf and 0 hld
pcisel_h    = (ptbe_l[] == B"0000")
            & (ptnum_h[] == rwCtrl);      -- (LCELL) compensate for clk buf and 0 hld

-- Output Bits
!selecttri_l.in     = select_h;
selecttri_l.oe      = !tsoe_l;
select_l            = selecttri_l.out;

!rdtri_l.in         = rd_h;
rdtri_l.oe          = !tsoe_l;
rd_l                = rdtri_l.out;

!wrtri_l.in         = wr_h;
wrtri_l.oe          = !tsoe_l;
wr_l                = wrtri_l.out;

!ptadrtri_l.in      = ptadr_h;
ptadrtri_l.oe       = !tsoe_l;
ptadr_l             = ptadrtri_l.out;

!ptrdytri_l.in      = ptrdy_h;
ptrdytri_l.oe       = !tsoe_l;
ptrdy_l             = ptrdytri_l.out;

!busgrnttri_l.in    = busgrnt_h;
busgrnttri_l.oe     = !tsoe_l;
pcibusgrnt_l        = busgrnttri_l.out;

%===============================================================
Pass Thru State Machine
===============================================================%

ptSM.clk    = clk_r;
ptSM.reset  = !clr_l;

-- Output Bits
--addonrdack_h.clk      = clk_r;
--!addonrdack_h.clrn    = !clr_l;

--addonwrack_h.clk      = clk_r;
--!addonwrack_h.clrn    = !clr_l;

-- The LCELL delay here is actually beneficial. It helps prevent dq_h[]
-- bus conflict during the time 'ptadr_h' is deactivating and the
-- write controller begins driving the dq_h[] bus with data.
```

```
!dqoe_l      = dqoe_h;        -- LCELL

CASE ptSM IS
    WHEN pt00 =>            -- <nothing> active
        IF pcibusgrnt_l THEN
            IF ptatn_h THEN
                IF pcisel_h THEN ptSM = pt01;
                END IF;
            ELSIF addonwr_h THEN ptSM = pt09;
            ELSIF addonrd_h THEN ptSM = pt11;
            ELSIF addonint_h THEN ptSM = pt07;
            END IF;
        END IF;

    -- Get Passthru Address
    WHEN pt01 =>            -- busgrnt_h, select_h, ptadr_h, tsoe_l active
        ptSM = pt02;
    WHEN pt02 =>            -- busgrnt_h, select_h, ptadr_h, tsoe_l active
        IF ptwr_h THEN ptSM = pt03;
        ELSE ptSM = pt05;
        END IF;

    -- Passthru Write Operations
    WHEN pt03 =>            -- busgrnt_h, select_h, rd_h, tsoe_l active
        IF !ptburst_h & !ptatn_h THEN ptSM = pt12;
        ELSE ptSM = pt04;
        END IF;
    WHEN pt04 =>            -- busgrnt_h, select_h, rd_h, ptrdy_h, tsoe_l active
        IF ptatn_h THEN ptSM = pt03;
        END IF;

    -- Passthru Read Operations
    WHEN pt05 =>            -- busgrnt_h, select_h, wr_h, dqoe_h, tsoe_l active
        IF !ptburst_h & !ptatn_h THEN ptSM = pt12;
        ELSE ptSM = pt06;
        END IF;
    WHEN pt06 =>            -- busgrnt_h, select_h, wr_h, ptrdy_h, dqoe_h, tsoe_l active
        IF ptatn_h THEN ptSM = pt05;
        END IF;

    -- Addon Interrupt Operations
    WHEN pt07 =>            -- busgrnt_h, dqoe_h, tsoe_l, addonintack_h active
        IF doaddonint_h THEN ptSM = pt08;
        ELSIF !addonint_h THEN ptSM = pt12;
        END IF;
    WHEN pt08 =>            -- busgrnt_h, select_h, wr_h, dqoe_h, tsoe_l, addonintack_h active
        IF !addonint_h THEN ptSM = pt12;
        ELSE ptSM = pt07;
        END IF;

    -- Addon Write Operations
    WHEN pt09 =>            -- busgrnt_h, dqoe_h, tsoe_l, addonwrack_h active
        IF doaddonwr_h THEN ptSM = pt10;
        ELSIF !addonwr_h THEN ptSM = pt12;
        END IF;
    WHEN pt10 =>            -- busgrnt_h, select_h, wr_h, dqoe_h, tsoe_l, addonwrack_h active
        IF !addonwr_h THEN ptSM = pt12;
        ELSE ptSM = pt09;
        END IF;

    -- Addon Read Operations
    WHEN pt11 =>            -- busgrnt_h, select_h, rd_h, tsoe_l, addonrdack_h active
        IF !addonrd_h THEN ptSM = pt12;
        END IF;
```

```
    -- Drive Control Signals Inactive for One Clock
    WHEN pt12 =>        -- tsoe_l active
        ptSM = pt00;
END CASE;

%================================================================
Read/Write State Machine
================================================================%
rwSM.clk    = clk_r;
rwSM.reset  = !clr_l;

pcilinecnttcdly_h.clk   = clk_r;
!pcilinecnttcdly_h.clrn = !clr_l;
pcilinecnttcdly_h.d     = pcilinecnttc_h;

go_h = !pcilinecnttc_h & pcilinecnttcdly_h;

-- Valid Interrupt Counter
-- Due to a race condition during master mode read transfers, we must
-- wait a "few" clock cycles after irq_l asserts to determine whether
-- or not the AMCC's read FIFO is empty. If so, then we really are done
-- with the transfer. Master mode write transfers are done as soon as
-- irq_l is detected asserted.
irqcnt_h[].clk      = clk_r;
!irqcnt_h[].clrn    = !clr_l;
IF !irq_l & rdempty_h THEN
    irqcnt_h[].d = irqcnt_h[].q + 1;
ELSE
    irqcnt_h[].d = 0;
END IF;
IF write_h THEN
    irqcnttc_h = !irq_l;                    -- LCELL
ELSE
    irqcnttc_h = (irqcnt_h[].q == H"3");    -- LCELL
END IF;

-- Output Bits
addonrd_h.clk   = clk_r;
!addonrd_h.clrn = !clr_l;

addonwr_h.clk   = clk_r;
!addonwr_h.clrn = !clr_l;

CASE rwSM IS
    WHEN rw00 =>        -- <nothing> active
        IF go_h THEN rwSM = rw01; addonwr_h.s = VCC;
        END IF;
    WHEN rw01 =>        -- (addonwr_h) active
        IF addonwrack_h THEN rwSM = rw02;
        END IF;

    -- First, enable transfer count for bus master transfers
    WHEN rw02 =>        -- AGCSTS_h, doaddonwr_h (addonwr_h) active
        rwSM = rw03;
    WHEN rw03 =>        -- AGCSTS_h (addonwr_h) active
        rwSM = rw04;

    -- Next, clear any previous interrupts and enable future ones
    WHEN rw04 =>        -- AINT_h, doaddonwr_h (addonwr_h) active
        rwSM = rw05;
    WHEN rw05 =>        -- AINT_h (addonwr_h) active
        rwSM = rw06;
```

```
    -- Next, write PCI Address Register to either MWAR or MRAR
    WHEN rw06 =>          -- adrreg_h, doaddonwr_h (addonwr_h) active
        rwSM = rw07;
    WHEN rw07 =>          -- adrreg_h (addonwr_h) active
        rwSM = rw08;

    -- Next, write PCI Pixel Register to either MWTC or MRTC
    WHEN rw08 =>          -- tcreg_h, doaddonwr_h (addonwr_h) active
        rwSM = rw09;
    WHEN rw09 =>          -- tcreg_h (addonwr_h) active
        addonwr_h.r = VCC;
        rwSM = rw10;

    -- Check that 'irq_l' has gone inactive (as a result of writing to AINT)
    WHEN rw10 =>          -- (checkirq_h) active
        checkirq_h = VCC;
        IF !irq_l # (!write_h & !rdempty_h) THEN rwSM = rw00;   -- abort with irqerr_h!
        ELSE rwSM = rw11;
        END IF;

    -- Now, assert amwen_h or amren_h and wait for irq_l
    WHEN rw11 =>          -- (xfren_h) active
        xfren_h = VCC;
        IF irqcnttc_h THEN rwSM = rw12; addonrd_h.s = VCC;
        END IF;
    WHEN rw12 =>          -- (addonrd_h) active
        IF addonrdack_h THEN rwSM = rw13;
        END IF;

    -- Read AINT register and trap on errors
    WHEN rw13 =>          -- AINT_h (addonrd_h) active
        rwSM = rw14;
    WHEN rw14 =>          -- AINT_h (addonrd_h, checkints_h) active
        checkints_h = VCC;
        addonrd_h.r = VCC;
        IF dq_h[20] # dq_h[21] THEN rwSM = rw00;     -- bistint_h or mtabort_h!
        ELSE rwSM = rw15;
        END IF;

    -- Decrement PCI Line Counter/Register. If transitioning to zero, we're done
    WHEN rw15 =>          -- decrline_h active
        IF done_h THEN rwSM = rw00;
        ELSE rwSM = rw16; addonrd_h.s = VCC;
        END IF;
    WHEN rw16 =>          -- (addonrd_h) active
        IF addonrdack_h THEN rwSM = rw17;
        END IF;

    -- Read either MWAR or MRAR into PCI Address Register
    WHEN rw17 =>          -- adrreg_h (addonrd_h) active
        rwSM = rw18;
    WHEN rw18 =>          -- adrreg_h (addonrd_h, loadpciadrreg_h) active
        loadpciadrreg_h = VCC;
        addonrd_h.r = VCC;
        rwSM = rw19;

    -- Add PCI Address Increment Register
    WHEN rw19 =>          -- <nothing> active (allow add to ripple through)
        rwSM = rw20;
    WHEN rw20 =>          -- (add_h) active
        add_h = VCC;
        addonwr_h.s = VCC;
        rwSM = rw01;
END CASE;
```

```
%=======================================================================
Mail Box State Machine
=======================================================================%
mbSM.clk     = clk_r;
mbSM.reset   = !clr_l;

CASE mbSM IS
    -- ==== Interrupts not currently active ====
    WHEN mb00 =>          -- mbena_h active (enables mbcnt_h[] AND mbff_h[])
        mbSM = mb01;
    WHEN mb01 =>          -- <nothing> active
        IF mbff_h[].q != B"00" THEN mbSM = mb02;
        ELSE mbSM = mb00;
        END IF;

    -- ==== Interrupts currently active ====
    -- Generate an interrupt
    WHEN mb02 =>          -- addonint_h active
        IF addonintack_h THEN mbSM = mb03;
        END IF;
    WHEN mb03 =>          -- addonint_h, AOMB4_h, doaddonint_h  active
        mbSM = mb04;
    WHEN mb04 =>          -- addonint_h, AOMB4_h active
        mbSM = mb05;

    -- Loop awhile, checking to see if interrupts are acknowledged
    WHEN mb05 =>          -- <nothing> active
        IF mbff_h[].q == B"00" THEN mbSM = mb00;
        ELSE mbSM = mb06;
        END IF;
    WHEN mb06 =>          -- mbena_h active
        IF mbcnttc_h THEN mbSM = mb02;
        ELSE mbSM = mb05;
        END IF;
END CASE;

mbff_h[].clk      = clk_r;
!mbff_h[].clrn    = !clr_l;
mbff_h[].ena      = mbena_h;
mbff_h[1].d       = rwinten_h & rwint_h;
mbff_h[0].d       = wrint_h;

emb_h[]           = 0;
embclk_r          = GND;
%
emb_h[7..2]       = 0;
emb_h[1..0]       = mbff_h[1..0].q;
%
mbcnt_h[].clk     = clk_r;
!mbcnt_h[].clrn = !clr_l;
mbcnt_h[].ena     = mbena_h;
IF (mbSM == mb00) THEN
    mbcnt_h[].d = 0;
ELSE
    mbcnt_h[].d = mbcnt_h[].q + 1;
END IF;
mbcnttc_h         = (mbcnt_h[].q == H"7");

%=======================================================================
Write Operations
=======================================================================%
wrsel_h               = rd_h & ptrdy_h & ptatn_h;      -- "rd_h & ptrdy_h" is same as 'pt04'!
```

```
ctrlwr_h          = wrsel_h & (ptaddr_h[] == pciCtrlStat);     -- LCELL
pcitcregwr_h      = wrsel_h & (ptaddr_h[] == pciTCReg);        -- LCELL
pciadrincwr_h     = wrsel_h & (ptaddr_h[] == pciAdrInc);       -- LCELL
pciadrregwr_h     = wrsel_h & (ptaddr_h[] == pciAdrReg) # loadpciadrreg_h;     -- LCELL
pcilinecntwr_h    = wrsel_h & (ptaddr_h[] == pciLineCnt);      -- LCELL

-- Control/Status Register Bits
color_h[].clk     = clk_r;
!color_h[].clrn   = !clr_l;
color_h[].ena     = ctrlwr_h;
color_h[].d       = dq_h[01..00];

depth_h[].clk     = clk_r;
!depth_h[].clrn   = !clr_l;
depth_h[].ena     = ctrlwr_h;
depth_h[].d       = dq_h[03..02];

write_h.clk       = clk_r;
!write_h.clrn     = !clr_l;
write_h.ena       = ctrlwr_h;
write_h.d         = dq_h[04];

rwint_h           = irqerr_h # bistint_h # mtabort_h # dmadone_h; -- LCELL

rwinten_h.clk     = clk_r;
!rwinten_h.clrn   = !clr_l;
rwinten_h.ena     = ctrlwr_h;
rwinten_h.d       = dq_h[06];

irqerr_h.clk      = clk_r;
!irqerr_h.clrn    = !clr_l;
irqerr_h.r        = irqerr_h.q & ctrlwr_h & !dq_h[07]
                  & !(checkirq_h & (!irq_l # !rdempty_h));
irqerr_h.s        = checkirq_h & (!irq_l # !rdempty_h);

bistint_h.clk     = clk_r;
!bistint_h.clrn   = !clr_l;
bistint_h.r       = bistint_h.q & ctrlwr_h & !dq_h[08]
                  & !(checkints_h & dq_h[20]);
bistint_h.s       = checkints_h & dq_h[20];

mtabort_h.clk     = clk_r;
!mtabort_h.clrn   = !clr_l;
mtabort_h.r       = mtabort_h.q & ctrlwr_h & !dq_h[09]
                  & !(checkints_h & dq_h[21]);
mtabort_h.s       = checkints_h & dq_h[21];

dmadone_h.clk     = clk_r;
!dmadone_h.clrn   = !clr_l;
dmadone_h.r       = dmadone_h.q & ctrlwr_h & !dq_h[10]
                  & !done_h;
dmadone_h.s       = done_h;

%===================================================================
Read Operations
===================================================================%
-- We add an additional LCELL buffer here (ie dqbuf_h[]) in order
-- to minimize fan-in to dqtri_h[]. It costs 32 LCELLs but I think
-- the reduction in interrconnect resources is worth it.
CASE ptaddr_h[] IS
    WHEN pciCtrlStat =>
        dqbuf_h[31..11] = GND;
        dqbuf_h[10]     = dmadone_h;
        dqbuf_h[09]     = mtabort_h;
```

```
            dqbuf_h[08]      = bistint_h;
            dqbuf_h[07]      = irqerr_h;
            dqbuf_h[06]      = rwinten_h;
            dqbuf_h[05]      = rwint_h;
            dqbuf_h[04]      = write_h;
            dqbuf_h[03..02] = depth_h[1..0];
            dqbuf_h[01..00] = color_h[1..0];
        WHEN pciTCReg =>
            dqbuf_h[31..15] = GND;
            dqbuf_h[14..02] = pcitcreg_h[14..02];
            dqbuf_h[01..00] = GND;
        WHEN pciAdrInc =>
            dqbuf_h[31..14] = GND;
            dqbuf_h[13..02] = pciadrinc_h[13..02];
            dqbuf_h[01..00] = GND;
        WHEN pciAdrReg =>
            dqbuf_h[31..02] = pciadrreg_h[31..02];
            dqbuf_h[01..00] = GND;
-- The following wasn't in the original design when the pinouts were frozen.
-- It can go back in when the pins are allowed to float.
--      WHEN pciLineCnt =>
--          dqbuf_h[31..11] = GND;
--          dqbuf_h[10..00] = pcilinecnt_h[10..00];
    END CASE;

    -- For writes to 'AINT_h', use H"003F4000" for master mode writes
    --                          use H"003F8000" for master mode reads
    -- AGCSTS_h uses H"10000000"
    -- which enables transfer count registers
    IF AGCSTS_h THEN
        dqtri_h[31..00].in = H"10000000";
    ELSIF AINT_h THEN
        dqtri_h[31..16].in = H"003F";
        IF write_h THEN dqtri_h[15..14].in = B"01";
        ELSE dqtri_h[15..14].in = B"10";
        END IF;
        dqtri_h[13..00].in = GND;
    ELSIF AOMB4_h THEN
        dqtri_h[31..02].in = 0;
        dqtri_h[01..00].in = mbff_h[];       -- Mailbox #4, byte #1 (#3 is unavailable)
    ELSIF adrreg_h THEN
        dqtri_h[31..02].in = pciadrreg_h[31..02];
        dqtri_h[01..00].in = GND;
    ELSIF tcreg_h THEN
        dqtri_h[31..15].in = GND;
        dqtri_h[14..02].in = pcitcreg_h[14..02];
        dqtri_h[01..00].in = GND;
    ELSE
        dqtri_h[31..00].in = dqbuf_h[31..00];
    END IF;

    dqtri_h[].oe     = !dqoe_l;
    dq_h[]           = dqtri_h[].out;

    %==================================================================
    PCI Transfer Counter Register (pcitcreg_h[14..02])
    ==================================================================%

    pcitcreg_h[].clk       = clk_r;
    !pcitcreg_h[].clrn     = !clr_l;
    CASE pcitcregwr_h IS
        WHEN B"0" =>         -- hold last value
            pcitcreg_h[14..02].d = pcitcreg_h[14..02].q;
        WHEN B"1" =>         -- load register
            pcitcreg_h[14..02].d = dq_h[14..02];
```

```
END CASE;

%==========================================================================
PCI Address Increment Register (pciadrinc_h[13..02])
==========================================================================%

pciadrinc_h[].clk       = clk_r;
!pciadrinc_h[].clrn     = !clr_l;
CASE pciadrincwr_h IS
    WHEN B"0" =>
        pciadrinc_h[13..02].d = pciadrinc_h[13..02].q;
    WHEN B"1" =>
        pciadrinc_h[13..02].d = dq_h[13..02];
END CASE;

%==========================================================================
PCI Address Register (pciadrreg_h[31..02])
==========================================================================%

pciadrreg_h[].clk       = clk_r;
!pciadrreg_h[].clrn     = !clr_l;

CASE (pciadrregwr_h, add_h) IS
    WHEN B"01" =>          -- increment register by pciadrinc[]
        pciadrreg_h[31..02].d = sum_h[31..02];
    WHEN B"10" =>          -- load register/counter
        pciadrreg_h[31..02].d = dq_h[31..02];
    WHEN OTHERS =>         -- hold last value
        pciadrreg_h[31..02].d = pciadrreg_h[31..02].q;
END CASE;

-- Sum Register (Note: carry1_h & carry2_h are declared as LCELLs)
(carry1_h, sum_h[11..02])   = (B"0", pciadrreg_h[11..02].q) + (B"0", pciadrinc_h[11..02].q);
(carry2_h, sum_h[21..12])   = (B"0", pciadrreg_h[21..12].q) + (B"000000000", pciadrinc_h[13..12].q)
                            + (B"0000000000", carry1_h);
sum_h[31..22]               = pciadrreg_h[31..22].q + (B"000000000", carry2_h);

%==========================================================================
PCI Line Counter/Register (pcilinecnt_h[10..00])
==========================================================================%

pcilinecnt_h[].clk      = clk_r;
!pcilinecnt_h[].clrn    = !clr_l;

CASE (pcilinecntwr_h, decrline_h) IS
    WHEN B"01" =>          -- decrement counter
        pcilinecnt_h[10..00].d = pcilinecnt_h[10..00].q - 1;
    WHEN B"10" =>          -- load counter/register
        pcilinecnt_h[10..00].d = dq_h[10..00];
    WHEN OTHERS =>         -- hold last value
        pcilinecnt_h[10..00].d = pcilinecnt_h[10..00].q;
END CASE;

done_h          = LCELL (decrline_h & (pcilinecnt_h[] == 1));
pcilinecnttc_h  = LCELL (pcilinecnt_h[] == 0);

%==========================================================================
'adr_h[]' Multiplexer
==========================================================================%

adrtri_h[].oe   = !tsoe_l;
adr_h[]         = adrtri_h[].out;
IF AGCSTS_h THEN
    adrtri_h[].in = AGCSTS;
ELSIF AINT_h THEN
    adrtri_h[].in = AINT;
ELSIF ACMB4_h THEN
    adrtri_h[].in = ACMB4;
```

```
    ELSIF adrreg_h THEN
        IF write_h THEN
            adrtri_h[].in = MWAR;
        ELSE
            adrtri_h[].in = MRAR;
        END IF;
    ELSIF tcreg_h THEN
        IF write_h THEN
            adrtri_h[].in = MWTC;
        ELSE
            adrtri_h[].in = MRTC;
        END IF;
    ELSE
        adrtri_h[].in = APTD;
    END IF;

    %==========================================================================
    'amwen_h' and 'amren_h' Multiplexer
    ==========================================================================%
    IF xfren_h THEN
        IF write_h THEN amwen_h = VCC;
        ELSE amren_h = VCC;
        END IF;
    END IF;

END;
```

```
%=============================================================================
RDWRCTLR.TDF - Frame Capture Board Read/Write Controller
Copyright (c) 1996, Kensal Corporation

Revision History:
1.00    K.W. Crocker    9 Jul 96
    Initial writing.
=============================================================================%
TITLE "Frame Capture Board Read/Write Controller";

-- Longword offset within passthru region
CONSTANT pciCtrlStat    = H"0";
CONSTANT pciPixReg      = H"1";
CONSTANT pciAdrInc      = H"2";
CONSTANT pciAdrReg      = H"3";
CONSTANT pciLineCnt     = H"4";
CONSTANT rwPixReg       = H"5";
CONSTANT rwPixCnt       = H"6"; -- Read only
CONSTANT rwAdrInc       = H"7";
CONSTANT rwAdrCnt       = H"8";
CONSTANT rdLineCnt      = H"9";

-- Values for depth_h[1..0]
CONSTANT video16        = H"0";
CONSTANT video32        = H"1";
CONSTANT data8          = H"2";
CONSTANT rsvdDepth      = H"3";

-- Values for color_h[1..0]
CONSTANT monored        = H"0";
CONSTANT monogrn        = H"1";
CONSTANT monoblu        = H"2";
CONSTANT rgb            = H"3";

-- Values for rwmuxsel_h[4..0]
CONSTANT rwmslsred32    = H"00";    -- video mode, LS monochrome red, 32-bit, read only
CONSTANT rwmsmsred32    = H"01";    -- video mode, MS monochrome red, 32-bit, read only
CONSTANT rwmslsgrn32    = H"02";    -- video mode, LS monochrome green, 32-bit, read only
CONSTANT rwmsmsgrn32    = H"03";    -- video mode, MS monochrome green, 32-bit, read only
CONSTANT rwmslsblu32    = H"04";    -- video mode, LS monochrome blue, 32-bit, read only
CONSTANT rwmsmsblu32    = H"05";    -- video mode, MS monochrome blue, 32-bit, read only
CONSTANT rwmslsrgb32    = H"06";    -- video mode, LS RGB, 32-bit
CONSTANT rwmsmsrgb32    = H"07";    -- video mode, MS RGB, 32-bit
CONSTANT rwmsred16      = H"08";    -- video mode, monochrome red, 16-bit, read only
CONSTANT rwmsgrn16      = H"09";    -- video mode, monochrome green, 16-bit, read only
CONSTANT rwmsblu16      = H"0A";    -- video mode, monochrome blue, 16-bit, read only
CONSTANT rwmsrgb16      = H"0B";    -- video mode, RGB, 16-bit, read only
CONSTANT rwmslsred8     = H"10";    -- data mode, LS monochrome red, 8-bit
CONSTANT rwmsmsred8     = H"11";    -- data mode, MS monochrome red, 8-bit
CONSTANT rwmslsgrn8     = H"12";    -- data mode, LS monochrome green, 8-bit
CONSTANT rwmsmsgrn8     = H"13";    -- data mode, MS monochrome green, 8-bit
CONSTANT rwmslsblu8     = H"14";    -- data mode, LS monochrome blue, 8-bit
CONSTANT rwmsmsblu8     = H"15";    -- data mode, MS monochrome blue, 8-bit

SUBDESIGN rdwrctlr (
    -- Clock and Asynchronous Reset Inputs
    bpclk_r             : INPUT;    -- 33 MHz buffered PCI clock (global)
    sysrst_l            : INPUT;    -- clear from PCI controller (global)

    -- AMCC S5933 Interface Signals
    dq_h[20..00]        : INPUT;    -- data bus
    wrfifo_l            : OUTPUT;   -- write fifo strobe
    rdfifo_l            : OUTPUT;   -- read fifo strobe
    wrfull_h            : INPUT;    -- write fifo full input
```

```
    rdempty_h            : INPUT;    -- read fifo empty input

    -- Passthru Interface Signals
    dqout_h[31..00]      : OUTPUT;
    dqbusreq_h           : INPUT;    -- inverted (with LCELL) version of 'ptatn_l'
    ptaddr_h[3..0]       : INPUT;
    wrsel_h              : INPUT;
    busgrnt_l            : BIDIR;

    -- Interface with Super Mux
    rwmuxsel_h[4..0]     : OUTPUT;
    rwa_h[9..0]          : OUTPUT;
    rwdoe_h              : OUTPUT;
    rwras_h              : OUTPUT;
    rwcas_h              : OUTPUT;
    rwallras_h           : OUTPUT;
    rwallcas_h           : OUTPUT;
    rwwe_h               : OUTPUT;
    rwor_h               : INPUT;    -- Super Mux FIFO's output ready
    rwren_h              : OUTPUT;   -- Super Mux FIFO's read enable

    -- Interface with Altera PCI Controller
    depth_h[1..0]        : INPUT;    -- depth_h[] field from control register
    color_h[1..0]        : INPUT;    -- color_h[] field from control register

    -- Diagnostic I/O
    rfshforce_h          : INPUT;
)

VARIABLE
    clk_r                : NODE;
    clr_l                : NODE;

    -- AMCC S5933 Interface Signals
    dqbuf_h[31..00]      : LCELL;
    busgrnttri_l         : TRI;

    -- Refresh Counter
    rfshcnt_h[8..0]      : DFF;      -- Refresh counter
    rfshreq_h            : SRFF;

    -- Read Line Counter/Register
    rdlinecnt_h[10..00]  : DFF;      -- Loaded by CPU, decremented every line to zero
    rdlinecntdecr_h      : DFF;
    rdlinecntwr_h        : NODE;
    rdlinecnttc_h        : LCELL;

    -- Read/Write Pixel Register
    rwpixreg_h[12..00]   : DFF;      -- DRAM read-writes/video line
    rwpixregwr_h         : NODE;

    -- Read/Write Pixel Counter/Register
    rwpixcnt_h[12..00]   : DFF;      -- Loads rwpixreg_h[], then decrements to zero.
    rwpixcnttc_h         : LCELL;
    rwpixcntdecr_h       : DFF;
    qload_h              : SRFF;
    endline_h            : SRFF;

    -- Read/Write Address Increment Register
    rwadrinc_h[20..00]   : DFF;      -- Added to rwadrcnt_h[] at end of line.
    rwadrincwr_h         : NODE;

    -- Read/Write Address Counter/Register
    rwadrcnt_h[20..00]   : DFF;      -- Generates DRAM address and LSB/MSB select
```

```
rwadrcntwr_h        : NODE;
countbytwo_h        : LCELL;
carry_h             : LCELL;    -- Carry out on [01..00]
endpage_h           : LCELL;    -- Carry out on [10..00]

-- Summation Register
sum_h[20..00]       : LCELL;    -- Sum of read/write adr cntr and read/write adr incr

-- CBR State Machine
cbrdone_h           : NODE;
cbrSM               : MACHINE OF BITS (
                        rwallcas_h, rwallras_h
                      ) WITH STATES (
                        cIdle       = B"00",
                        cCBR1       = B"10",
                        cCBR2       = B"11",
                        cCBR3       = B"01"
                      );

-- Register Load State Machine
regldSM             : MACHINE OF BITS (
                        load_h, add_h
                      ) WITH STATES (
                        rlIdle      = B"00",
                        rlLoad      = B"10",
                        rlWait      = B"00",
                        rlAdd       = B"11"
                      );

-- DRAM State Machine
qcbr_h              : NODE;
qadd_h              : NODE;
pcLastCAS           : LCELL;
reading_h           : SRFF;
dramSM              : MACHINE OF BITS (
                        busgrnt_h, busgrntoe_l, rdfifo_h, rwras_h,
                        rwcas_h, colsel_h, rwwe_h, count_h
                      ) WITH STATES (
                        dIdle       = B"01000000",
                        dRfsh       = B"01000000",
                        dRasPre     = B"01000000",

                        dWrRas      = B"10010000",
                        dWrColSel1  = B"10110110",
                        dWrCas1     = B"10011111",
                        dWrColSel2  = B"10010110",
                        dWrCas2     = B"10011110",
                        dWrCas3     = B"10011111",

                        dRdRas      = B"10010000",
                        dRdColSel1  = B"10010100",
                        dRdCas1     = B"10011101",
                        dRdColSel2  = B"10010100",
                        dRdCas2     = B"10011101",
                        dRdDone1    = B"10010000",
                        dRdDone2    = B"10000000",

                        dRdFifo     = B"10000000",

                        dDone       = B"00000000",
                        dNewLn      = B"01000000"
                      );

readSM              : MACHINE OF BITS (
```

```
                    rwdoe_h, rwren_h, wrfifo_h
                ) WITH STATES (
                    rIdle       = B"000",
                    rArmed      = B"100",
                    rRead       = B"111",
                    rWait1      = B"100",
                    rWait2      = B"100"
                );

BEGIN
    clk_r           = GLOBAL (bpclk_r);
    clr_l           = GLOBAL (sysrst_l);

    %-----------------------------------------------------------------------
    PCI Controller Interface
    -----------------------------------------------------------------------%
    CASE ptaddr_h[] IS
        WHEN rwPixReg =>
            dqbuf_h[31..13] = GND;
            dqbuf_h[12..00] = rwpixreg_h[12..00];
        WHEN rwPixCnt =>
            dqbuf_h[31..13] = GND;
            dqbuf_h[12..00] = rwpixcnt_h[12..00];
        WHEN rwAdrInc =>
            dqbuf_h[31..21] = GND;
            dqbuf_h[20..00] = rwadrinc_h[20..00];
        WHEN rwAdrCnt =>
            dqbuf_h[31..21] = GND;
            dqbuf_h[20..00] = rwadrcnt_h[20..00];
        WHEN rdLineCnt =>
            dqbuf_h[31..11] = GND;
            dqbuf_h[10..00] = rdlinecnt_h[10..00];
    END CASE;
    dqout_h[31..00] = dqbuf_h[31..00];

    rwpixregwr_h    = LCELL (wrsel_h & (ptaddr_h[] == rwPixReg));
    rwadrincwr_h    = LCELL (wrsel_h & (ptaddr_h[] == rwAdrInc));
    rwadrcntwr_h    = LCELL (wrsel_h & (ptaddr_h[] == rwAdrCnt));
    rdlinecntwr_h   = LCELL (wrsel_h & (ptaddr_h[] == rdLineCnt));

    !wrfifo_l       = wrfifo_h;
    !rdfifo_l       = rdfifo_h;

    %-----------------------------------------------------------------------
    Read Line Counter/Register (rdlinecnt_h[10..00])

    This register is used to determine when read operations (ie. FIFO to PCI
    writes) should occur. When the register is loaded with a non-zero value,
    DRAM reads will be started, throttled by 'wrfull_h', and ending when
    the register is decremented to zero.
    -----------------------------------------------------------------------%
    rdlinecnt_h[].clk   = clk_r;
    !rdlinecnt_h[].clrn = !clr_l;

    CASE (rdlinecntwr_h, rdlinecntdecr_h) IS
        WHEN B"01" =>           -- decrement counter
            rdlinecnt_h[10..00].d = rdlinecnt_h[10..00].q - 1;
        WHEN B"10" =>           -- load counter/register
            rdlinecnt_h[10..00].d = dq_h[10..00];
        WHEN OTHERS =>          -- hold last value
            rdlinecnt_h[10..00].d = rdlinecnt_h[10..00].q;
    END CASE;

    rdlinecnttc_h           = (rdlinecnt_h[10..00] == 0);       -- LCELL
```

```
rdlinecntdecr_h.clk       = clk_r;
!rdlinecntdecr_h.clrn     = !clr_l;
rdlinecntdecr_h.d         = pcLastCAS & !rdlinecnttc_h & rwpixcnttc_h;

%---------------------------------------------------------------------
Read/Write Pixel Register (rwpixreg_h[12..00])

The Pixel Register is a zero-based register. For example, for a line with
ten pixels on it, load a value of 0x9.
---------------------------------------------------------------------%
rwpixreg_h[].clk    = clk_r;
!rwpixreg_h[].clrn  = !clr_l;
CASE rwpixregwr_h IS
    WHEN B"0" =>         -- hold last value
        rwpixreg_h[].d = rwpixreg_h[].q;
    WHEN B"1" =>         -- load register
        rwpixreg_h[].d = dq_h[12..00];
END CASE;

%---------------------------------------------------------------------
Read/Write Pixel Counter/Register (rwpixcnt_h[12..00])

This register is read only from the PCI bus
---------------------------------------------------------------------%
rwpixcnt_h[].clk    = clk_r;
!rwpixcnt_h[].clrn  = !clr_l;
CASE (load_h, rwpixcntdecr_h) IS
    WHEN B"01" =>        -- decrement counter
        rwpixcnt_h[].d = rwpixcnt_h[].q - 1;
    WHEN B"10" =>        -- load counter/register
        rwpixcnt_h[].d = rwpixreg_h[].q;
    WHEN OTHERS =>       -- hold last value
        rwpixcnt_h[].d = rwpixcnt_h[].q;
END CASE;
rwpixcnttc_h    = (rwpixcnt_h[] == 0);   -- LCELL

rwpixcntdecr_h.clk       = clk_r;
!rwpixcntdecr_h.clrn     = !clr_l;
rwpixcntdecr_h.d         = pcLastCAS & !rwpixcnttc_h;

-- To automatically force a load after 'rwpixreg_h[]' is loaded,
-- we set 'qload_h'.
qload_h.clk      = clk_r;
!qload_h.clrn    = !clr_l;
qload_h.s        = rwpixregwr_h;
qload_h.r        = load_h;

-- 'endline_h' goes active the cycle 'count_h' is active,
-- and stays active until it is reset (aka endline_h.r = VCC)
endline_h.clk    = clk_r;
!endline_h.clrn = !clr_l;
endline_h.s      = pcLastCAS & rwpixcnttc_h;

%---------------------------------------------------------------------
Read/Write Address Increment Register (rwadrinc_h[20..00])
---------------------------------------------------------------------%
rwadrinc_h[].clk    = clk_r;
!rwadrinc_h[].clrn  = !clr_l;
CASE rwadrincwr_h IS
    WHEN B"0" =>         -- hold last value
        rwadrinc_h[].d = rwadrinc_h[].q;
    WHEN B"1" =>         -- load register
        rwadrinc_h[].d = dq_h[20..00];
```

```
END CASE;

%------------------------------------------------------------------------
Read/Write Address Counter/Register (rwadrcnt_h[20..00])

Note that only the 20 MS bits go the the DRAM (multiplexed 10-bits each).
The LS bit selects between LS and MS bytes of the 16-bit DRAM word.
------------------------------------------------------------------------%
rwadrcnt_h[].clk    = clk_r;
!rwadrcnt_h[].clrn  = !clr_l;

-- Read/Write Address Counter/Register (part 1)
CASE (rwadrcntwr_h, add_h, count_h, countbytwo_h) IS
    WHEN B"0010" =>       -- increment counter by 1
        rwadrcnt_h[00].d = !rwadrcnt_h[00].q;
    WHEN B"0011" =>       -- increment counter by 2
        rwadrcnt_h[00].d = rwadrcnt_h[00].q;
    WHEN B"010X" =>       -- increment register by sum_h[]
        rwadrcnt_h[00].d = sum_h[00];
    WHEN B"100X" =>       -- load register
        rwadrcnt_h[00].d = dq_h[00];
    WHEN OTHERS =>        -- hold last value
        rwadrcnt_h[00].d = rwadrcnt_h[00].q;
END CASE;
carry_h     = count_h & (!countbytwo_h & rwadrcnt_h[00] # countbytwo_h);    -- LCELL

-- Read/Write Address Counter/Register (part 2)
CASE (rwadrcntwr_h, add_h, carry_h) IS
    WHEN B"001" =>        -- increment counter by 1
        rwadrcnt_h[10..01].d = rwadrcnt_h[10..01].q + 1;
    WHEN B"010" =>        -- increment register by sum_h[]
        rwadrcnt_h[10..01].d = sum_h[10..01];
    WHEN B"100" =>        -- load register
        rwadrcnt_h[10..01].d = dq_h[10..01];
    WHEN OTHERS =>        -- hold last value
        rwadrcnt_h[10..01].d = rwadrcnt_h[10..01].q;
END CASE;

-- Since 'endpage_h' goes active the cycle 'count_h' is active,
-- it will only be active for one clock because the counter
-- will rollover to a non-terminal count.
endpage_h    = count_h & !countbytwo_h & (rwadrcnt_h[10..00] == B"11111111111")
             # count_h &  countbytwo_h & (rwadrcnt_h[10..01] == B"1111111111");  -- LCELL

-- Read/Write Address Counter/Register (part 3)
CASE (rwadrcntwr_h, add_h, endpage_h) IS
    WHEN B"001" =>        -- increment counter by 1
        rwadrcnt_h[20..11].d = rwadrcnt_h[20..11].q + 1;
    WHEN B"010" =>        -- increment register by sum_h[]
        rwadrcnt_h[20..11].d = sum_h[20..11];
    WHEN B"100" =>        -- load register
        rwadrcnt_h[20..11].d = dq_h[20..11];
    WHEN OTHERS =>        -- hold last value
        rwadrcnt_h[20..11].d = rwadrcnt_h[20..11].q;
END CASE;

-- Sum Register
sum_h[] = rwadrinc_h[].q + rwadrcnt_h[].q;

%------------------------------------------------------------------------
DRAM Refresh Counter

Refreshes will be scheduled every 1/2**9 (512) clocks.
The clock period is 30 ns (33.3330 MHz), so this
```

```
equates to 15.360 us/CBR. At that rate, the entire
1,024 rows will be refreshed in 15.729 ms. The
specification for the part is 16 ms.
---------------------------------------------------------------------------%
rfshcnt_h[].clk      = clk_r;
!rfshcnt_h[].clrn    = !clr_l;
rfshcnt_h[].d        = rfshcnt_h[].q + 1;

rfshreq_h.clk        = clk_r;
!rfshreq_h.clrn      = !clr_l;
rfshreq_h.s          = rfshcnt_h[] == 2 # rfshforce_h;
rfshreq_h.r          = cbrdone_h;

%---------------------------------------------------------------------------
CBR DRAM Refresh State Machine
---------------------------------------------------------------------------%
cbrSM.clk            = clk_r;
cbrSM.reset          = !clr_l;

-- Output Bits
cbrdone_h    = SOFT (cCBR3);

CASE cbrSM IS
    WHEN cIdle =>             -- <nothing> active
        IF qcbr_h THEN cbrSM = cCBR1;
        END IF;
    WHEN cCBR1 =>             -- allcas_h active
        cbrSM = cCBR2;
    WHEN cCBR2 =>             -- allcas_h, allras_h active
        cbrSM = cCBR3;
    WHEN cCBR3 =>             -- allras_h, cbrdone_h active
        cbrSM = cIdle;        -- rfshreq_h inactive on next cycle
END CASE;

%---------------------------------------------------------------------------
Register Load State Machine

This machine controls the loading of the rwpixcnt_h[] and rwadrcnt_h[]
registers.
---------------------------------------------------------------------------%
regldSM.clk      = clk_r;
regldSM.reset    = !clr_l;

CASE regldSM IS
    WHEN rlIdle =>            -- <nothing> active
        IF qload_h THEN regldSM = rlLoad;
        ELSIF qadd_h THEN regldSM = rlWait;
        END IF;

    WHEN rlLoad =>            -- load_h active
        regldSM = rlIdle;

    WHEN rlWait =>            -- <nothing> active
        regldSM = rlAdd;
    WHEN rlAdd =>             -- load_h, add_h active
        regldSM = rlIdle;
END CASE;

%---------------------------------------------------------------------------
DRAM Read/Write State Machine
---------------------------------------------------------------------------%
dramSM.clk       = clk_r;
dramSM.reset     = !clr_l;
```

```
-- pcLastCAS Precursor Bit. Goes active before entering state dWrCas3 or dRdCas2
pcLastCAS    = dRdColSel1 & !wrfull_h & LCELL (depth_h[] != data8) # dRdColSel2
             # dWrCas2 # dWrColSel2; -- LCELL

!busgrnttri_l.in     = busgrnt_h;
busgrnttri_l.oe      = !busgrntoe_l;
!busgrnt_l           = !busgrnttri_l.out;

-- 'reading_h' is an S/R FF that is active when the dramSM is reading DRAM. It
-- signals the readSM that it should be ready (ie. armed) to read from the SuperMUX's
-- FIFO and transfer the data to the S5933Q's FIFO.
reading_h.clk        = clk_r;
!reading_h.clrn      = !clr_l;

CASE dramSM IS
    WHEN dIdle =>            -- <nothing> active
        IF rfshreq_h.q THEN dramSM = dRfsh;
        ELSIF busgrnt_l & !dqbusreq_h THEN
            IF !rdempty_h THEN dramSM = dWrRas;
            ELSIF !wrfull_h & !rdlinecnttc_h THEN dramSM = dRdRas; reading_h.s = VCC;
            ELSIF !wrfull_h & rwor_h THEN dramSM = dRdFifo; reading_h.s = VCC;
            END IF;
        END IF;
    WHEN dRfsh =>            -- (qcbr_h) active
        qcbr_h = VCC;
        IF cbrdone_h THEN dramSM = dRasPre;
        END IF;
    WHEN dRasPre =>          -- <nothing> active
        dramSM = dIdle;

    WHEN dWrRas =>           -- busgrnt_h, busgrntoe_l, rwras_h active
        dramSM = dWrColSel1;
    WHEN dWrColSel1 =>       -- busgrnt_h, busgrntoe_l, rdfifo_h, rwras_h, rwwe_h, colsel_h active
        IF depth_h[] == data8 THEN dramSM = dWrCas1;
        ELSE dramSM = dWrCas2;
        END IF;
    WHEN dWrCas1 =>          -- busgrnt_h, busgrntoe_l, rwras_h, rwcas_h, colsel_h, rwwe_h, count_h
active
        dramSM = dWrColSel2;
    WHEN dWrColSel2 =>       -- busgrnt_h, busgrntoe_l, rwras_h, colsel_h, rwwe_h active
        dramSM = dWrCas3;
    -- Allow time for 'rdempty_h' to settle
    WHEN dWrCas2 =>          -- busgrnt_h, busgrntoe_l, rwras_h, rwcas_h, colsel_h, rwwe_h active
        dramSM = dWrCas3;
    WHEN dWrCas3 =>          -- busgrnt_h, busgrntoe_l, rwras_h, rwcas_h, colsel_h, rwwe_h, count_h
active
        IF !endline_h.q & !rfshreq_h.q & !rdempty_h & !endpage_h & !dqbusreq_h THEN dramSM =
dWrColSel1;
        ELSE dramSM = dDone;
        END IF;

    WHEN dRdRas =>           -- busgrnt_h, busgrntoe_l, rwras_h (reading_h) active
        dramSM = dRdColSel1;
    WHEN dRdColSel1 =>       -- busgrnt_h, busgrntoe_l, rwras_h, colsel_h  (reading_h) active
        IF wrfull_h THEN dramSM = dRdDone1;
        ELSIF depth_h[] == data8 THEN dramSM = dRdCas1;
        ELSE dramSM = dRdCas2;
        END IF;
    WHEN dRdCas1 =>          -- busgrnt_h, busgrntoe_l, rwras_h, rwcas_h, colsel_h, count_h
 (reading_h) active
        dramSM = dRdColSel2;
    WHEN dRdColSel2 =>       -- busgrnt_h, busgrntoe_l, rwras_h, colsel_h (reading_h) active
        dramSM = dRdCas2;
```

```
      WHEN dRdCas2 =>          -- busgrnt_h, busgrntoe_l, rwras_h, rwcas_h, colsel_h, count_h
(reading_h) active
          IF !endline_h.q & !rfshreq_h.q & !endpage_h & !dqbusreq_h THEN dramSM = dRdColSel1;
          ELSE dramSM = dRdDone1;
          END IF;
      WHEN dRdDone1 =>         -- busgrnt_h, busgrntoe_l, rwras_h (reading_h) active
          dramSM = dRdDone2;
      WHEN dRdDone2 =>         -- busgrnt_h, busgrntoe_l active
          reading_h.r = VCC;
          IF !reading_h.q & (readSM == rIdle) THEN dramSM = dDone;
          END IF;

      WHEN dRdFifo =>          -- busgrnt_h, busgrntoe_l (reading_h) active
          IF rfshreq_h.q # dqbusreq_h # !rwor_h THEN dramSM = dRdDone2;
          END IF;

      -- Drive control signals inactive one clock
      WHEN dDone =>            -- busgrntoe_l active
          IF endline_h.q THEN dramSM = dNewLn;
          ELSIF rfshreq_h.q THEN dramSM = dRfsh;
          ELSE dramSM = dIdle;
          END IF;
      WHEN dNewLn =>           -- (qadd_h) active
          qadd_h = VCC;
          endline_h.r = VCC;
          IF add_h THEN dramSM = dIdle;
          END IF;
  END CASE;

  %-----------------------------------------------------------------------
  SuperMux Read/S5933Q FIFO Write State Machine
  -----------------------------------------------------------------------%
  readSM.clk      = clk_r;
  readSM.reset    = !clr_l;

  CASE readSM IS
      WHEN rIdle =>            -- <nothing> active
          IF reading_h THEN readSM = rArmed;
          END IF;
      WHEN rArmed =>           -- rwdoe_h active
          IF rwor_h & !wrfull_h THEN readSM = rRead;
          ELSIF !reading_h THEN
              IF wrfull_h THEN readSM = rIdle;     -- give up immediately!
              ELSE readSM = rWait1;    -- !reading_h & !wrfull_h & !rwor_h
              END IF;
          END IF;

      WHEN rRead =>            -- rwdoe_h, rwren_h, wrfifo_h active
          readSM = rArmed;

      -- We've received a request to stop transfers ('!reading_h'), possibly
      -- because we've finished doing DRAM reads. The S5933Q's FIFO isn't full,
      -- but we don't have anything to send (perhaps because it hasn't gone
      -- through the SMUX pipeline yet. We wait for a couple of clocks for the
      -- SMUX's output FIFO to fill before giving up.
      WHEN rWait1 =>           -- <nothing> active
          IF rwor_h THEN readSM = rArmed;
          ELSE readSM = rWait2;
          END IF;
      WHEN rWait2 =>           -- <nothing> active
          IF rwor_h THEN readSM = rArmed;
          ELSE readSM = rIdle;
          END IF;
  END CASE;
```

```
%-----------------------------------------------------------------------
Video DRAM Address Multiplexor
-----------------------------------------------------------------------%
CASE colsel_h IS
    WHEN B"0" =>
        rwa_h[] = rwadrcnt_h[20..11].q;
    WHEN B"1" =>
        rwa_h[] = rwadrcnt_h[10..01].q;
END CASE;

%-----------------------------------------------------------------------
Super Mux Select Signals
-----------------------------------------------------------------------%
CASE depth_h[1..0] IS
    -- requires 2 DRAM cycles (same addr) per 2 32-bit writes (counts as 2 pixels)
    WHEN video32 =>
        rwmuxsel_h[4..3] = B"00";
        rwmuxsel_h[0] = rwadrcnt_h[00];
        CASE color_h[1..0] IS
            WHEN monored =>
                rwmuxsel_h[2..1] = B"00";
            WHEN monogrn =>
                rwmuxsel_h[2..1] = B"01";
            WHEN monoblu =>
                rwmuxsel_h[2..1] = B"10";
            WHEN rgb =>
                rwmuxsel_h[2..1] = B"11";
        END CASE;
    -- requires 1 DRAM cycle (unique addr) per 32-bit write (counts as 2 pixels)
    WHEN video16 =>
        countbytwo_h = VCC;
        rwmuxsel_h[4..2] = B"010";
        CASE color_h[1..0] IS
            WHEN monored =>
                rwmuxsel_h[1..0] = B"00";
            WHEN monogrn =>
                rwmuxsel_h[1..0] = B"01";
            WHEN monoblu =>
                rwmuxsel_h[1..0] = B"10";
            WHEN rgb =>
                rwmuxsel_h[1..0] = B"11";
        END CASE;
    -- requires 2 DRAM cycles (unique addr) per 32-bit write (counts as 1 pixel)
    WHEN data8 =>
        countbytwo_h = VCC;
        rwmuxsel_h[4..3] = B"10";
        rwmuxsel_h[0] = rwadrcnt_h[01];
        CASE color_h[1..0] IS
            WHEN monored =>
                rwmuxsel_h[2..1] = B"00";
            WHEN monogrn =>
                rwmuxsel_h[2..1] = B"01";
            WHEN monoblu =>
                rwmuxsel_h[2..1] = B"10";
        END CASE;
    END CASE;

END; % RDWRCTLR.TDF %
```

```
%==========================================================================
FIFO3.TDF - Frame Capture Board FIFO
Copyright (c) 1996, Kensal Corporation

Revision History:
1.00    K.W. Crocker    3 Jul 96
    Initial writing. Writes to MSbyte advance the FIFO.
==========================================================================%
TITLE "Frame Capture Board FIFO";

SUBDESIGN fifo3 (
    clk_r               : INPUT;
    clr_l               : INPUT;

    din_h[31..00]       : INPUT;
    dout_h[31..00]      : OUTPUT;

    rwsel_h             : INPUT;
    wenls_h             : INPUT;
    wenms_h             : INPUT;
    or_h                : OUTPUT;
    ren_h               : INPUT;

    fifoerr_h           : OUTPUT;
)

VARIABLE
    rega_h[31..00]      : DFFE;
    regb_h[31..00]      : DFFE;
    regc_h[31..00]      : DFFE;
    regaenms_h          : LCELL;
    regaenls_h          : LCELL;
    regbenms_h          : LCELL;
    regbenls_h          : LCELL;
    regcenms_h          : LCELL;
    regcenls_h          : LCELL;
    btoams_h            : LCELL;
    btoals_h            : LCELL;
    ctobms_h            : LCELL;
    ctobls_h            : LCELL;

    fSM                 : MACHINE OF BITS (
                            empty_h, one_h, two_h, three_h, or_h, fifoerr_h
                        ) WITH STATES (
                            fEmpty  = B"100000",
                            fOne    = B"010010",
                            fTwo    = B"001010",
                            fThree  = B"000110",
                            fErr    = B"000001"
                        );

BEGIN
    rega_h[].clk        = clk_r;
    !rega_h[].clrn      = !clr_l;
    IF btoams_h THEN
        rega_h[31..16].d = regb_h[31..16];
    ELSE
        rega_h[31..16].d = din_h[31..16];
    END IF;
    IF btoals_h THEN
        rega_h[15..00].d = regb_h[15..00];
    ELSE
        rega_h[15..00].d = din_h[15..00];
    END IF;
```

```
    dout_h[]     = rega_h[].q;

    regb_h[].clk        = clk_r;
    !regb_h[].clrn      = !clr_l;
    IF ctobms_h THEN
        regb_h[31..16].d = regc_h[31..16];
    ELSE
        regb_h[31..16].d = din_h[31..16];
    END IF;
    IF ctobls_h THEN
        regb_h[15..00].d = regc_h[15..00];
    ELSE
        regb_h[15..00].d = din_h[15..00];
    END IF;

    regc_h[].clk        = clk_r;
    !regc_h[].clrn      = !clr_l;
    regc_h[].d          = din_h[];

    btoams_h     = two_h & ren_h # three_h & ren_h;                               -- LCELL
    btoals_h     = one_h & ren_h & !wenls_h # two_h & ren_h # three_h & ren_h;    -- LCELL
    ctobms_h     = three_h & ren_h;                                               -- LCELL
    ctobls_h     = two_h & ren_h & !wenls_h # three_h & ren_h;                    -- LCELL

    regaenms_h   = empty_h & wenms_h # one_h & ren_h & wenms_h # two_h & ren_h # three_h & ren_h;      --
LCELL
    regaenls_h   = empty_h & wenls_h # one_h & ren_h # two_h & ren_h # three_h & ren_h;  -- LCELL
    regbenms_h   = one_h & !ren_h & wenms_h # two_h & ren_h & wenms_h # three_h & ren_h; -- LCELL
    regbenls_h   = one_h & !ren_h & wenls_h # two_h & ren_h # three_h & ren_h;     -- LCELL
    regcenms_h   = two_h & !ren_h & wenms_h # three_h & ren_h & wenms_h; -- LCELL
    regcenls_h   = two_h & !ren_h & wenls_h # three_h & ren_h & wenls_h; -- LCELL
    rega_h[31..16].ena = regaenms_h;
    rega_h[15..00].ena = regaenls_h;
    regb_h[31..16].ena = regbenms_h;
    regb_h[15..00].ena = regbenls_h;
    regc_h[31..16].ena = regcenms_h;
    regc_h[15..00].ena = regcenls_h;

    fSM.clk      = clk_r;
    fSM.reset    = !clr_l;

    CASE fSM IS
        WHEN fEmpty =>        -- empty_h active
            IF rwsel_h & ren_h THEN fSM = fErr;
            ELSIF wenms_h THEN fSM = fOne;
            END IF;
        WHEN fOne =>          -- one_h, or_h active
            IF wenms_h & !ren_h THEN fSM = fTwo;
            ELSIF !wenms_h & rwsel_h & ren_h THEN fSM = fEmpty;
            END IF;
        WHEN fTwo =>          -- two_h, or_h active
            IF wenms_h & !ren_h THEN fSM = fThree;
            ELSIF !wenms_h & rwsel_h & ren_h THEN fSM = fOne;
            END IF;
        WHEN fThree =>        -- three_h, or_h active
            IF wenms_h & !ren_h THEN fSM = fErr;
            ELSIF !wenms_h & rwsel_h & ren_h THEN fSM = fTwo;
            END IF;
        WHEN fErr =>          -- fifoerr_h active
            fSM = fErr;
    END CASE;
END;
```

```
%=====================================================================
SMUX.TDF - Frame Capture Board Super Mux
Copyright (c) 1996, Kensal Corporation

Revision History:
1.00    K.W. Crocker    18 Jun 96
    Initial writing.
1.01    K.W. Crocker    21 Jun 96
    Divided responsibility for handling each DRAM bank to a separate chip to
    reduce I/O requirements. Both chips are identical.
1.02    K.W. Crocker    12 Dec 97
    Swapped MS and LS words for 16-bit reads to conform to pixel '0' being
    in MSW and pixel '1' being in LSW.
=====================================================================%
TITLE "Frame Capture Board Super Mux";

INCLUDE "fifo3.inc";

-- Values for wrmuxsel_h[2..0]
CONSTANT wrmsred        = H"0";     -- monochrome red
CONSTANT wrmsgrn        = H"1";     -- monochrome green
CONSTANT wrmsblu        = H"2";     -- monochrome blue
CONSTANT wrmsrsrvd      = H"3";     -- reserved
CONSTANT wrmslsredgrn   = H"4";     -- RGB LS red/grn
CONSTANT wrmslsblu      = H"5";     -- RGB LS blue
CONSTANT wrmsmsredgrn   = H"6";     -- RGB MS red/grn
CONSTANT wrmsmsblu      = H"7";     -- RGB MS blue

-- Values for rwmuxsel_h[4..0]
CONSTANT rwmslsred32    = H"00";    -- video mode, LS monochrome red, 32-bit, read only
CONSTANT rwmsmsred32    = H"01";    -- video mode, MS monochrome red, 32-bit, read only
CONSTANT rwmslsgrn32    = H"02";    -- video mode, LS monochrome green, 32-bit, read only
CONSTANT rwmsmsgrn32    = H"03";    -- video mode, MS monochrome green, 32-bit, read only
CONSTANT rwmslsblu32    = H"04";    -- video mode, LS monochrome blue, 32-bit, read only
CONSTANT rwmsmsblu32    = H"05";    -- video mode, MS monochrome blue, 32-bit, read only
CONSTANT rwmslsrgb32    = H"06";    -- video mode, LS RGB, 32-bit
CONSTANT rwmsmsrgb32    = H"07";    -- video mode, MS RGB, 32-bit
CONSTANT rwmsred16      = H"08";    -- video mode, monochrome red, 16-bit, read only
CONSTANT rwmsgrn16      = H"09";    -- video mode, monochrome green, 16-bit, read only
CONSTANT rwmsblu16      = H"0A";    -- video mode, monochrome blue, 16-bit, read only
CONSTANT rwmsrgb16      = H"0B";    -- video mode, RGB, 16-bit, read only
CONSTANT rwmslsred8     = H"10";    -- data mode, LS monochrome red, 8-bit
CONSTANT rwmsmsred8     = H"11";    -- data mode, MS monochrome red, 8-bit
CONSTANT rwmslsgrn8     = H"12";    -- data mode, LS monochrome green, 8-bit
CONSTANT rwmsmsgrn8     = H"13";    -- data mode, MS monochrome green, 8-bit
CONSTANT rwmslsblu8     = H"14";    -- data mode, LS monochrome blue, 8-bit
CONSTANT rwmsmsblu8     = H"15";    -- data mode, MS monochrome blue, 8-bit

SUBDESIGN smux (                        -- 226 I/O pins req'd (to handle both banks)
                                        -- 158 I/O pins to handle one bank

                                        -- 174 I/O pins to handle both banks, write only
                                        -- 192 I/O pins to handle both banks, read/write only
    -- Clock and Asynchronous Reset Inputs
    36mhz_r             : INPUT;        -- 36 MHz crystal input (global)
    bpclk_r             : INPUT;        -- 33 MHz buffered PCI clock (global)
    gclr_l              : INPUT;        -- clear from PCI controller (global)

    -- Interface with Write Controller
    wrmuxsel_h[2..0]    : INPUT;
    wra_h[9..0]         : INPUT;
    wrd_h[15..00]       : INPUT;
    wrras_h             : INPUT;
    wrcas_h             : INPUT;
```

```
    wrallras_h          : INPUT;
    wrallcas_h          : INPUT;
    wrwe_h              : INPUT;

    -- Interface with Read/Write Controller
    rwmuxsel_h[4..0]    : INPUT;
    rwa_h[9..0]         : INPUT;
    rwd_h[31..00]       : BIDIR;
    rwdoe_h             : INPUT;
    rwras_h             : INPUT;
    rwcas_h             : INPUT;
    rwallras_h          : INPUT;
    rwallcas_h          : INPUT;
    rwwe_h              : INPUT;
    rwor_h              : OUTPUT;
    rwren_h             : INPUT;
    rdfifo_l            : INPUT;

    -- Video DRAM Bank Select Bit (Dynamic)
    writebank1_h        : INPUT;        -- when active, Write Controller is writing to bank #1

    -- Bank Select Bit (Static)
    ctrlbank1_h         : INPUT;        -- H -> this controller is connected to bank #1
                                        -- L -> this controller is connected to bank #0
    -- Video DRAM Interface
    rd_h[15..00]        : BIDIR;
    gd_h[15..00]        : BIDIR;
    bd_h[15..00]        : BIDIR;
    a_h[9..0]           : OUTPUT;       -- slow slew rate
    rras_l              : OUTPUT;
    rcasl_l             : OUTPUT;
    rcash_l             : OUTPUT;
    gras_l              : OUTPUT;
    gcasl_l             : OUTPUT;
    gcash_l             : OUTPUT;
    bras_l              : OUTPUT;
    bcasl_l             : OUTPUT;
    bcash_l             : OUTPUT;
    we_l                : OUTPUT;

    -- Diagnostic Outputs
    fifoerr_h           : OUTPUT;
    wenlsout_h          : OUTPUT;
    wenmsout_h          : OUTPUT;
    or_h                : OUTPUT;
    rwselout_h          : OUTPUT;
)

VARIABLE
    wrclk_r             : NODE;
    rwclk_r             : NODE;
    clr_l               : NODE;

    rwsel_h             : LCELL;    -- H -> RWCtlr, L -> WrCtlr connected

    wrmuxselin_h[2..0]  : DFF;      -- input IOCELL F/F
    wrain_h[9..0]       : DFF;      -- input IOCELL F/F
    wrdin_h[15..00]     : DFF;      -- input IOCELL F/F
    wrrasin_h           : DFF;      -- input IOCELL F/F
    wrcasin_h           : DFF;      -- input IOCELL F/F
    wrallrasin_h        : DFF;      -- input IOCELL F/F
    wrallcasin_h        : DFF;      -- input IOCELL F/F
    wrwein_h            : DFF;      -- input IOCELL F/F
```

```
    rwmuxselin_h[4..0]    : DFF;        -- input IOCELL F/F
    rwmuxseld1_h[4..0]    : DFF;
    rwmuxseld2_h[4..0]    : DFF;
    rwain_h[9..0]         : DFF;        -- input IOCELL F/F
    rwdin_h[31..00]       : DFFE;       -- input IOCELL F/F
    rwdout_h[31..00]      : LCELL;
    rwdtrioe_l            : LCELL;
    rwdtri_h[31..00]      : TRI;
    rwrasin_h             : DFF;        -- input IOCELL F/F
    rwcasin_h             : DFF;        -- input IOCELL F/F
    rwallrasin_h          : DFF;        -- input IOCELL F/F
    rwallcasin_h          : DFF;        -- input IOCELL F/F
    rwwein_h              : DFF;        -- input IOCELL F/F
    enams_h               : NODE;
    enals_h               : NODE;
    rwseld1_h             : LCELL;
    rwseld2_h             : LCELL;
    rwortri_h             : TRI;
    rworoe_l              : LCELL;

    rdtri_h[15..00]       : TRI;
    gdtri_h[15..00]       : TRI;
    bdtri_h[15..00]       : TRI;
    rdtrioe_l             : LCELL;
    gdtrioe_l             : LCELL;
    bdtrioe_l             : LCELL;

    wenms1_h              : DFF;
    wenms2_h              : DFF;
    wenls1_h              : DFF;
    wenls2_h              : DFF;
    rwfifo                : fifo3;

    busycnt_h[1..0]       : DFF;
    idle_h                : LCELL;

BEGIN
--  wrclk_r      = 36mhz_r;
--  rwclk_r      = bpclk_r;
--  clr_l        = gclr_l;
    wrclk_r      = GLOBAL (36mhz_r);
    rwclk_r      = GLOBAL (bpclk_r);
    clr_l        = GLOBAL (gclr_l);

    -- 'rwsel_h' = 1 -> RWCtrl connected, 'rwsel_h' = 0 -> WrCtlr connected
    rwsel_h      = ctrlbank1_h $ writebank1_h;     -- LCELL

    rwselout_h   = rwsel_h;        -- DIAGNOSTIC
    wenmsout_h   = wenms2_h.q;     -- DIAGNOSTIC
    wenlsout_h   = wenls2_h.q;     -- DIAGNOSTIC

    wrmuxselin_h[].clk    = wrclk_r;
    wrmuxselin_h[].d      = wrmuxsel_h[];
    wrain_h[].clk         = wrclk_r;
    wrain_h[].d           = wra_h[];
    wrdin_h[].clk         = wrclk_r;
    wrdin_h[].d           = wrd_h[];
    wrrasin_h.clk         = 36mhz_r;       -- non-global clock for delay
    wrrasin_h.d           = wrras_h;
    wrcasin_h.clk         = 36mhz_r;       -- non-global clock for delay
    wrcasin_h.d           = wrcas_h;
    wrallrasin_h.clk      = wrclk_r;
    wrallrasin_h.d        = wrallras_h;
    wrallcasin_h.clk      = wrclk_r;
```

```
wrallcasin_h.d          = wrallcas_h;
wrwein_h.clk            = wrclk_r;
wrwein_h.d              = wrwe_h;

rwmuxselin_h[].clk      = rwclk_r;
rwmuxselin_h[].d        = rwmuxsel_h[];
rwain_h[].clk           = rwclk_r;
rwain_h[].d             = rwa_h[];
rwdin_h[].clk           = rwclk_r;
rwdin_h[].ena           = !rdfifo_l;
rwdin_h[].d             = rwd_h[];
rwrasin_h.clk           = bpclk_r;      -- non-global clock for delay
rwrasin_h.d             = rwras_h;
rwcasin_h.clk           = bpclk_r;      -- non-global clock for delay
rwcasin_h.d             = rwcas_h;
rwallrasin_h.clk        = rwclk_r;
rwallrasin_h.d          = rwallras_h;
rwallcasin_h.clk        = rwclk_r;
rwallcasin_h.d          = rwallcas_h;
rwwein_h.clk            = rwclk_r;
rwwein_h.d              = rwwe_h;

rd_h[]                  = rdtri_h[].out;
gd_h[]                  = gdtri_h[].out;
bd_h[]                  = bdtri_h[].out;
rdtri_h[].oe            = !rdtrioe_l;
gdtri_h[].oe            = !gdtrioe_l;
bdtri_h[].oe            = !bdtrioe_l;

wenms1_h.clk            = rwclk_r;
!wenms1_h.clrn          = !clr_l;
wenms1_h.d              = !rwwein_h & rwcasin_h.q & enams_h;

wenms2_h.clk            = rwclk_r;
!wenms2_h.clrn          = !clr_l;
wenms2_h.d              = wenms1_h.q;

wenls1_h.clk            = rwclk_r;
!wenls1_h.clrn          = !clr_l;
wenls1_h.d              = !rwwein_h & rwcasin_h.q & enals_h;

wenls2_h.clk            = rwclk_r;
!wenls2_h.clrn          = !clr_l;
wenls2_h.d              = wenls1_h.q;

rwfifo.(clk_r, clr_l, rwsel_h, wenls_h, wenms_h, din_h[31..00], ren_h) =
    (rwclk_r, clr_l, rwsel_h, wenls2_h.q, wenms2_h.q, rwdout_h[], rwren_h);

(or_h, rwdtri_h[].in, fifoerr_h) =
    rwfifo.(or_h, dout_h[31..00], fifoerr_h);

!rwdtrioe_l             = rwdoe_h & rwsel_h;     -- LCELL
rwdtri_h[].oe           = !rwdtrioe_l;
rwd_h[]                 = rwdtri_h[].out;

rwseld1_h               = rwsel_h;               -- LCELL
rwseld2_h               = rwseld1_h;             -- LCELL
!rworoe_l               = rwsel_h & rwseld2_h;   -- LCELL. adds 2 LCELL delay for OE turnon

rwortri_h.in            = or_h;
rwortri_h.oe            = !rworoe_l;
rwor_h                  = rwortri_h.out;

rwmuxseld1_h[].clk         = rwclk_r;
```

```
!rwmuxseld1_h[].clrn      = !clr_l;
rwmuxseld1_h[].d          = rwmuxselin_h[].q;

rwmuxseld2_h[].clk        = rwclk_r;
!rwmuxseld2_h[].clrn      = !clr_l;
rwmuxseld2_h[].d          = rwmuxseld1_h[].q;

CASE rwmuxseld2_h[] IS
    WHEN rwmslsred32 =>        -- video mode, LS monochrome red, 32-bit, read only
        rwdout_h[31..24]       = GND;
        rwdout_h[23..16]       = rd_h[07..00];
        rwdout_h[15..08]       = rd_h[07..00];
        rwdout_h[07..00]       = rd_h[07..00];
    WHEN rwmsmsred32 =>        -- video mode, MS monochrome red, 32-bit, read only
        rwdout_h[31..24]       = GND;
        rwdout_h[23..16]       = rd_h[15..08];
        rwdout_h[15..08]       = rd_h[15..08];
        rwdout_h[07..00]       = rd_h[15..08];
    WHEN rwmslsgrn32 =>        -- video mode, LS monochrome green, 32-bit, read only
        rwdout_h[31..24]       = GND;
        rwdout_h[23..16]       = gd_h[07..00];
        rwdout_h[15..08]       = gd_h[07..00];
        rwdout_h[07..00]       = gd_h[07..00];
    WHEN rwmsmsgrn32 =>        -- video mode, MS monochrome green, 32-bit, read only
        rwdout_h[31..24]       = GND;
        rwdout_h[23..16]       = gd_h[15..08];
        rwdout_h[15..08]       = gd_h[15..08];
        rwdout_h[07..00]       = gd_h[15..08];
    WHEN rwmslsblu32 =>        -- video mode, LS monochrome blue, 32-bit, read only
        rwdout_h[31..24]       = GND;
        rwdout_h[23..16]       = bd_h[07..00];
        rwdout_h[15..08]       = bd_h[07..00];
        rwdout_h[07..00]       = bd_h[07..00];
    WHEN rwmsmsblu32 =>        -- video mode, MS monochrome blue, 32-bit, read only
        rwdout_h[31..24]       = GND;
        rwdout_h[23..16]       = bd_h[15..08];
        rwdout_h[15..08]       = bd_h[15..08];
        rwdout_h[07..00]       = bd_h[15..08];
    WHEN rwmslsrgb32 =>        -- video mode, LS RGB, 32-bit
        rwdout_h[31..24]       = GND;
        rwdout_h[23..16]       = rd_h[07..00];
        rwdout_h[15..08]       = gd_h[07..00];
        rwdout_h[07..00]       = bd_h[07..00];
    WHEN rwmsmsrgb32 =>        -- video mode, MS RGB, 32-bit
        rwdout_h[31..24]       = GND;
        rwdout_h[23..16]       = rd_h[15..08];
        rwdout_h[15..08]       = gd_h[15..08];
        rwdout_h[07..00]       = bd_h[15..08];
    WHEN rwmsred16 =>          -- video mode, monochrome red, 16-bit, read only
        rwdout_h[31]           = GND;
        rwdout_h[30..26]       = rd_h[07..03];
        rwdout_h[25..21]       = rd_h[07..03];
        rwdout_h[20..16]       = rd_h[07..03];
        rwdout_h[15]           = GND;
        rwdout_h[14..10]       = rd_h[15..11];
        rwdout_h[09..05]       = rd_h[15..11];
        rwdout_h[04..00]       = rd_h[15..11];
    WHEN rwmsgrn16 =>          -- video mode, monochrome green, 16-bit, read only
        rwdout_h[31]           = GND;
        rwdout_h[30..26]       = gd_h[07..03];
        rwdout_h[25..21]       = gd_h[07..03];
        rwdout_h[20..16]       = gd_h[07..03];
        rwdout_h[15]           = GND;
        rwdout_h[14..10]       = gd_h[15..11];
```

```
            rwdout_h[09..05]     = gd_h[15..11];
            rwdout_h[04..00]     = gd_h[15..11];
        WHEN rwmsblu16 =>        -- video mode, monochrome blue, 16-bit, read only
            rwdout_h[31]         = GND;
            rwdout_h[30..26]     = bd_h[07..03];
            rwdout_h[25..21]     = bd_h[07..03];
            rwdout_h[20..16]     = bd_h[07..03];
            rwdout_h[15]         = GND;
            rwdout_h[14..10]     = bd_h[15..11];
            rwdout_h[09..05]     = bd_h[15..11];
            rwdout_h[04..00]     = bd_h[15..11];
        WHEN rwmsrgb16 =>        -- video mode, RGB, 16-bit, read only
            rwdout_h[31]         = GND;
            rwdout_h[30..26]     = rd_h[07..03];
            rwdout_h[25..21]     = gd_h[07..03];
            rwdout_h[20..16]     = bd_h[07..03];
            rwdout_h[15]         = GND;
            rwdout_h[14..10]     = rd_h[15..11];
            rwdout_h[09..05]     = gd_h[15..11];
            rwdout_h[04..00]     = bd_h[15..11];
        WHEN rwmslsred8 =>       -- data mode, LS monochrome red, 8-bit
            rwdout_h[15..08]     = rd_h[15..08];
            rwdout_h[07..00]     = rd_h[07..00];
        WHEN rwmsmsred8 =>       -- data mode, MS monochrome red, 8-bit
            rwdout_h[31..24]     = rd_h[15..08];
            rwdout_h[23..16]     = rd_h[07..00];
        WHEN rwmslsgrn8 =>       -- data mode, LS monochrome green, 8-bit
            rwdout_h[15..08]     = gd_h[15..08];
            rwdout_h[07..00]     = gd_h[07..00];
        WHEN rwmsmsgrn8 =>       -- data mode, MS monochrome green, 8-bit
            rwdout_h[31..24]     = gd_h[15..08];
            rwdout_h[23..16]     = gd_h[07..00];
        WHEN rwmslsblu8 =>       -- data mode, LS monochrome blue, 8-bit
            rwdout_h[15..08]     = bd_h[15..08];
            rwdout_h[07..00]     = bd_h[07..00];
        WHEN rwmsmsblu8 =>       -- data mode, MS monochrome blue, 8-bit
            rwdout_h[31..24]     = bd_h[15..08];
            rwdout_h[23..16]     = bd_h[07..00];
    END CASE;

    IF rwsel_h THEN
        a_h[]            = rwain_h[].q;
        !we_l            = rwwein_h.q;

        CASE rwmuxselin_h[] IS
            WHEN rwmslsrgb32 =>      -- video mode, LS RGB, 32-bit
                rdtri_h[07..00].in   = rwdin_h[23..16].q;
                gdtri_h[07..00].in   = rwdin_h[15..08].q;
                bdtri_h[07..00].in   = rwdin_h[07..00].q;
            WHEN rwmsmsrgb32 =>      -- video mode, MS RGB, 32-bit
                rdtri_h[15..08].in   = rwdin_h[23..16].q;
                gdtri_h[15..08].in   = rwdin_h[15..08].q;
                bdtri_h[15..08].in   = rwdin_h[07..00].q;
            WHEN rwmslsred8 =>       -- data mode, LS monochrome red, 8-bit
                rdtri_h[15..08].in   = rwdin_h[15..08].q;
                rdtri_h[07..00].in   = rwdin_h[07..00].q;
            WHEN rwmsmsred8 =>       -- data mode, MS monochrome red, 8-bit
                rdtri_h[15..08].in   = rwdin_h[31..24].q;
                rdtri_h[07..00].in   = rwdin_h[23..16].q;
            WHEN rwmslsgrn8 =>       -- data mode, LS monochrome green, 8-bit
                gdtri_h[15..08].in   = rwdin_h[15..08].q;
                gdtri_h[07..00].in   = rwdin_h[07..00].q;
            WHEN rwmsmsgrn8 =>       -- data mode, MS monochrome green, 8-bit
                gdtri_h[15..08].in   = rwdin_h[31..24].q;
```

```
            gdtri_h[07..00].in  = rwdin_h[23..16].q;
        WHEN rwmslsblu8 =>      -- data mode, LS monochrome blue, 8-bit
            bdtri_h[15..08].in  = rwdin_h[15..08].q;
            bdtri_h[07..00].in  = rwdin_h[07..00].q;
        WHEN rwmsmsblu8 =>      -- data mode, MS monochrome blue, 8-bit
            bdtri_h[15..08].in  = rwdin_h[31..24].q;
            bdtri_h[07..00].in  = rwdin_h[23..16].q;
    END CASE;

    CASE rwmuxseld1_h[] IS
        WHEN rwmslsred32 =>     -- video mode, LS monochrome red, 32-bit, read only
            !rdtrioe_l          = idle_h;
            !gdtrioe_l          = VCC;
            !bdtrioe_l          = VCC;
            !rras_l             = rwrasin_h.q # rwallrasin_h.q;
            !rcasl_l            = rwcasin_h.q # rwallcasin_h.q;
            !rcash_l            = rwallcasin_h.q;
            !gras_l             = rwallrasin_h.q;
            !gcasl_l            = rwallcasin_h.q;
            !gcash_l            = rwallcasin_h.q;
            !bras_l             = rwallrasin_h.q;
            !bcasl_l            = rwallcasin_h.q;
            !bcash_l            = rwallcasin_h.q;
            enams_h             = VCC;
            enals_h             = VCC;
--          rwdout_h[31..24]    = GND;
--          rwdout_h[23..16]    = rd_h[07..00];
--          rwdout_h[15..08]    = rd_h[07..00];
--          rwdout_h[07..00]    = rd_h[07..00];
        WHEN rwmsmsred32 =>     -- video mode, MS monochrome red, 32-bit, read only
            !rdtrioe_l          = idle_h;
            !gdtrioe_l          = VCC;
            !bdtrioe_l          = VCC;
            !rras_l             = rwrasin_h.q # rwallrasin_h.q;
            !rcasl_l            = rwallcasin_h.q;
            !rcash_l            = rwcasin_h.q # rwallcasin_h.q;
            !gras_l             = rwallrasin_h.q;
            !gcasl_l            = rwallcasin_h.q;
            !gcash_l            = rwallcasin_h.q;
            !bras_l             = rwallrasin_h.q;
            !bcasl_l            = rwallcasin_h.q;
            !bcash_l            = rwallcasin_h.q;
            enams_h             = VCC;
            enals_h             = VCC;
--          rwdout_h[31..24]    = GND;
--          rwdout_h[23..16]    = rd_h[15..08];
--          rwdout_h[15..08]    = rd_h[15..08];
--          rwdout_h[07..00]    = rd_h[15..08];
        WHEN rwmslsgrn32 =>     -- video mode, LS monochrome green, 32-bit, read only
            !rdtrioe_l          = VCC;
            !gdtrioe_l          = idle_h;
            !bdtrioe_l          = VCC;
            !rras_l             = rwallrasin_h.q;
            !rcasl_l            = rwallcasin_h.q;
            !rcash_l            = rwallcasin_h.q;
            !gras_l             = rwrasin_h.q # rwallrasin_h.q;
            !gcasl_l            = rwcasin_h.q # rwallcasin_h.q;
            !gcash_l            = rwallcasin_h.q;
            !bras_l             = rwallrasin_h.q;
            !bcasl_l            = rwallcasin_h.q;
            !bcash_l            = rwallcasin_h.q;
            enams_h             = VCC;
            enals_h             = VCC;
--          rwdout_h[31..24]    = GND;
```

```
--          rwdout_h[23..16]     = gd_h[07..00];
--          rwdout_h[15..08]     = gd_h[07..00];
--          rwdout_h[07..00]     = gd_h[07..00];
        WHEN rwmsmsgrn32 =>      -- video mode, MS monochrome green, 32-bit, read only
            !rdtrioe_l           = VCC;
            !gdtrioe_l           = idle_h;
            !bdtrioe_l           = VCC;
            !rras_l              = rwallrasin_h.q;
            !rcasl_l             = rwallcasin_h.q;
            !rcash_l             = rwallcasin_h.q;
            !gras_l              = rwrasin_h.q # rwallrasin_h.q;
            !gcasl_l             = rwallcasin_h.q;
            !gcash_l             = rwcasin_h.q # rwallcasin_h.q;
            !bras_l              = rwallrasin_h.q;
            !bcasl_l             = rwallcasin_h.q;
            !bcash_l             = rwallcasin_h.q;
            enams_h              = VCC;
            enals_h              = VCC;
--          rwdout_h[31..24]     = GND;
--          rwdout_h[23..16]     = gd_h[15..08];
--          rwdout_h[15..08]     = gd_h[15..08];
--          rwdout_h[07..00]     = gd_h[15..08];
        WHEN rwmslsblu32 =>      -- video mode, LS monochrome blue, 32-bit, read only
            !rdtrioe_l           = VCC;
            !gdtrioe_l           = VCC;
            !bdtrioe_l           = idle_h;
            !rras_l              = rwallrasin_h.q;
            !rcasl_l             = rwallcasin_h.q;
            !rcash_l             = rwallcasin_h.q;
            !gras_l              = rwallrasin_h.q;
            !gcasl_l             = rwallcasin_h.q;
            !gcash_l             = rwallcasin_h.q;
            !bras_l              = rwrasin_h.q # rwallrasin_h.q;
            !bcasl_l             = rwcasin_h.q # rwallcasin_h.q;
            !bcash_l             = rwallcasin_h.q;
            enams_h              = VCC;
            enals_h              = VCC;
--          rwdout_h[31..24]     = GND;
--          rwdout_h[23..16]     = bd_h[07..00];
--          rwdout_h[15..08]     = bd_h[07..00];
--          rwdout_h[07..00]     = bd_h[07..00];
        WHEN rwmsmsblu32 =>      -- video mode, MS monochrome blue, 32-bit, read only
            !rdtrioe_l           = VCC;
            !gdtrioe_l           = VCC;
            !bdtrioe_l           = idle_h;
            !rras_l              = rwallrasin_h.q;
            !rcasl_l             = rwallcasin_h.q;
            !rcash_l             = rwallcasin_h.q;
            !gras_l              = rwallrasin_h.q;
            !gcasl_l             = rwallcasin_h.q;
            !gcash_l             = rwallcasin_h.q;
            !bras_l              = rwrasin_h.q # rwallrasin_h.q;
            !bcasl_l             = rwallcasin_h.q;
            !bcash_l             = rwcasin_h.q # rwallcasin_h.q;
            enams_h              = VCC;
            enals_h              = VCC;
--          rwdout_h[31..24]     = GND;
--          rwdout_h[23..16]     = bd_h[15..08];
--          rwdout_h[15..08]     = bd_h[15..08];
--          rwdout_h[07..00]     = bd_h[15..08];
        WHEN rwmslsrgb32 =>      -- video mode, LS RGB, 32-bit
--          rdtri_h[07..00].in   = rwdin_h[23..16].q;
--          gdtri_h[07..00].in   = rwdin_h[15..08].q;
--          bdtri_h[07..00].in   = rwdin_h[07..00].q;
```

```
            !rdtrioe_l          = rwwein_h.q # idle_h;
            !gdtrioe_l          = rwwein_h.q # idle_h;
            !bdtrioe_l          = rwwein_h.q # idle_h;
            !rras_l             = rwrasin_h.q # rwallrasin_h.q;
            !rcasl_l            = rwcasin_h.q # rwallcasin_h.q;
            !rcash_l            = rwallcasin_h.q;
            !gras_l             = rwrasin_h.q # rwallrasin_h.q;
            !gcasl_l            = rwcasin_h.q # rwallcasin_h.q;
            !gcash_l            = rwallcasin_h.q;
            !bras_l             = rwrasin_h.q # rwallrasin_h.q;
            !bcasl_l            = rwcasin_h.q # rwallcasin_h.q;
            !bcash_l            = rwallcasin_h.q;
            enams_h             = VCC;
            enals_h             = VCC;
--          rwdout_h[31..24]    = GND;
--          rwdout_h[23..16]    = rd_h[07..00];
--          rwdout_h[15..08]    = gd_h[07..00];
--          rwdout_h[07..00]    = bd_h[07..00];
        WHEN rwmsmsrgb32 =>     -- video mode, MS RGB, 32-bit
--          rdtri_h[15..08].in  = rwdin_h[23..16].q;
--          gdtri_h[15..08].in  = rwdin_h[15..08].q;
--          bdtri_h[15..08].in  = rwdin_h[07..00].q;
            !rdtrioe_l          = rwwein_h.q # idle_h;
            !gdtrioe_l          = rwwein_h.q # idle_h;
            !bdtrioe_l          = rwwein_h.q # idle_h;
            !rras_l             = rwrasin_h.q # rwallrasin_h.q;
            !rcasl_l            = rwallcasin_h.q;
            !rcash_l            = rwcasin_h.q # rwallcasin_h.q;
            !gras_l             = rwrasin_h.q # rwallrasin_h.q;
            !gcasl_l            = rwallcasin_h.q;
            !gcash_l            = rwcasin_h.q # rwallcasin_h.q;
            !bras_l             = rwrasin_h.q # rwallrasin_h.q;
            !bcasl_l            = rwallcasin_h.q;
            !bcash_l            = rwcasin_h.q # rwallcasin_h.q;
            enams_h             = VCC;
            enals_h             = VCC;
--          rwdout_h[31..24]    = GND;
--          rwdout_h[23..16]    = rd_h[15..08];
--          rwdout_h[15..08]    = gd_h[15..08];
--          rwdout_h[07..00]    = bd_h[15..08];
        WHEN rwmsred16 =>       -- video mode, monochrome red, 16-bit, read only
            !rdtrioe_l          = idle_h;
            !gdtrioe_l          = VCC;
            !bdtrioe_l          = VCC;
            !rras_l             = rwrasin_h.q # rwallrasin_h.q;
            !rcasl_l            = rwcasin_h.q # rwallcasin_h.q;
            !rcash_l            = rwcasin_h.q # rwallcasin_h.q;
            !gras_l             = rwallrasin_h.q;
            !gcasl_l            = rwallcasin_h.q;
            !gcash_l            = rwallcasin_h.q;
            !bras_l             = rwallrasin_h.q;
            !bcasl_l            = rwallcasin_h.q;
            !bcash_l            = rwallcasin_h.q;
            enams_h             = VCC;
            enals_h             = VCC;
--          rwdout_h[31]        = GND;
--          rwdout_h[30..26]    = rd_h[15..11];
--          rwdout_h[25..21]    = rd_h[15..11];
--          rwdout_h[20..16]    = rd_h[15..11];
--          rwdout_h[15]        = GND;
--          rwdout_h[14..10]    = rd_h[07..03];
--          rwdout_h[09..05]    = rd_h[07..03];
--          rwdout_h[04..00]    = rd_h[07..03];
        WHEN rwmsgrn16 =>       -- video mode, monochrome green, 16-bit, read only
```

```
            !rdtrioe_l         = VCC;
            !gdtrioe_l         = idle_h;
            !bdtrioe_l         = VCC;
            !rras_l            = rwallrasin_h.q;
            !rcasl_l           = rwallcasin_h.q;
            !rcash_l           = rwallcasin_h.q;
            !gras_l            = rwrasin_h.q # rwallrasin_h.q;
            !gcasl_l           = rwcasin_h.q # rwallcasin_h.q;
            !gcash_l           = rwcasin_h.q # rwallcasin_h.q;
            !bras_l            = rwallrasin_h.q;
            !bcasl_l           = rwallcasin_h.q;
            !bcash_l           = rwallcasin_h.q;
            enams_h            = VCC;
            enals_h            = VCC;
--          rwdout_h[31]       = GND;
--          rwdout_h[30..26]   = gd_h[15..11];
--          rwdout_h[25..21]   = gd_h[15..11];
--          rwdout_h[20..16]   = gd_h[15..11];
--          rwdout_h[15]       = GND;
--          rwdout_h[14..10]   = gd_h[07..03];
--          rwdout_h[09..05]   = gd_h[07..03];
--          rwdout_h[04..00]   = gd_h[07..03];
        WHEN rwmsblu16 =>      -- video mode, monochrome blue, 16-bit, read only
            !rdtrioe_l         = VCC;
            !gdtrioe_l         = VCC;
            !bdtrioe_l         = idle_h;
            !rras_l            = rwallrasin_h.q;
            !rcasl_l           = rwallcasin_h.q;
            !rcash_l           = rwallcasin_h.q;
            !gras_l            = rwallrasin_h.q;
            !gcasl_l           = rwallcasin_h.q;
            !gcash_l           = rwallcasin_h.q;
            !bras_l            = rwrasin_h.q # rwallrasin_h.q;
            !bcasl_l           = rwcasin_h.q # rwallcasin_h.q;
            !bcash_l           = rwcasin_h.q # rwallcasin_h.q;
            enams_h            = VCC;
            enals_h            = VCC;
--          rwdout_h[31]       = GND;
--          rwdout_h[30..26]   = bd_h[15..11];
--          rwdout_h[25..21]   = bd_h[15..11];
--          rwdout_h[20..16]   = bd_h[15..11];
--          rwdout_h[15]       = GND;
--          rwdout_h[14..10]   = bd_h[07..03];
--          rwdout_h[09..05]   = bd_h[07..03];
--          rwdout_h[04..00]   = bd_h[07..03];
        WHEN rwmsrgb16 =>      -- video mode, RGB, 16-bit, read only
            !rdtrioe_l         = idle_h;
            !gdtrioe_l         = idle_h;
            !bdtrioe_l         = idle_h;
            !rras_l            = rwrasin_h.q # rwallrasin_h.q;
            !rcasl_l           = rwcasin_h.q # rwallcasin_h.q;
            !rcash_l           = rwcasin_h.q # rwallcasin_h.q;
            !gras_l            = rwrasin_h.q # rwallrasin_h.q;
            !gcasl_l           = rwcasin_h.q # rwallcasin_h.q;
            !gcash_l           = rwcasin_h.q # rwallcasin_h.q;
            !bras_l            = rwrasin_h.q # rwallrasin_h.q;
            !bcasl_l           = rwcasin_h.q # rwallcasin_h.q;
            !bcash_l           = rwcasin_h.q # rwallcasin_h.q;
            enams_h            = VCC;
            enals_h            = VCC;
--          rwdout_h[31]       = GND;
--          rwdout_h[30..26]   = rd_h[15..11];
--          rwdout_h[25..21]   = gd_h[15..11];
--          rwdout_h[20..16]   = bd_h[15..11];
```

```
--          rwdout_h[15]           = GND;
--          rwdout_h[14..10]       = rd_h[07..03];
--          rwdout_h[09..05]       = gd_h[07..03];
--          rwdout_h[04..00]       = bd_h[07..03];
        WHEN rwmslsred8 =>         -- data mode, LS monochrome red, 8-bit
--          rdtri_h[15..08].in     = rwdin_h[15..08].q;
--          rdtri_h[07..00].in     = rwdin_h[07..00].q;
            !rdtrioe_l             = rwwein_h.q # idle_h;
            !gdtrioe_l             = VCC;
            !bdtrioe_l             = VCC;
            !rras_l                = rwrasin_h.q # rwallrasin_h.q;
            !rcasl_l               = rwcasin_h.q # rwallcasin_h.q;
            !rcash_l               = rwcasin_h.q # rwallcasin_h.q;
            !gras_l                = rwallrasin_h.q;
            !gcasl_l               = rwallcasin_h.q;
            !gcash_l               = rwallcasin_h.q;
            !bras_l                = rwallrasin_h.q;
            !bcasl_l               = rwallcasin_h.q;
            !bcash_l               = rwallcasin_h.q;
            enals_h                = VCC;
--          rwdout_h[15..08]       = rd_h[15..08];
--          rwdout_h[07..00]       = rd_h[07..00];
        WHEN rwmsmsred8 =>         -- data mode, MS monochrome red, 8-bit
--          rdtri_h[15..08].in     = rwdin_h[31..24].q;
--          rdtri_h[07..00].in     = rwdin_h[23..16].q;
            !rdtrioe_l             = rwwein_h.q # idle_h;
            !gdtrioe_l             = VCC;
            !bdtrioe_l             = VCC;
            !rras_l                = rwrasin_h.q # rwallrasin_h.q;
            !rcasl_l               = rwcasin_h.q # rwallcasin_h.q;
            !rcash_l               = rwcasin_h.q # rwallcasin_h.q;
            !gras_l                = rwallrasin_h.q;
            !gcasl_l               = rwallcasin_h.q;
            !gcash_l               = rwallcasin_h.q;
            !bras_l                = rwallrasin_h.q;
            !bcasl_l               = rwallcasin_h.q;
            !bcash_l               = rwallcasin_h.q;
            enams_h                = VCC;
--          rwdout_h[31..24]       = rd_h[15..08];
--          rwdout_h[23..16]       = rd_h[07..00];
        WHEN rwmslsgrn8 =>         -- data mode, LS monochrome green, 8-bit
--          gdtri_h[15..08].in     = rwdin_h[15..08].q;
--          gdtri_h[07..00].in     = rwdin_h[07..00].q;
            !rdtrioe_l             = VCC;
            !gdtrioe_l             = rwwein_h.q # idle_h;
            !bdtrioe_l             = VCC;
            !rras_l                = rwallrasin_h.q;
            !rcasl_l               = rwallcasin_h.q;
            !rcash_l               = rwallcasin_h.q;
            !gras_l                = rwrasin_h.q # rwallrasin_h.q;
            !gcasl_l               = rwcasin_h.q # rwallcasin_h.q;
            !gcash_l               = rwcasin_h.q # rwallcasin_h.q;
            !bras_l                = rwallrasin_h.q;
            !bcasl_l               = rwallcasin_h.q;
            !bcash_l               = rwallcasin_h.q;
            enals_h                = VCC;
--          rwdout_h[15..08]       = gd_h[15..08];
--          rwdout_h[07..00]       = gd_h[07..00];
        WHEN rwmsmsgrn8 =>         -- data mode, MS monochrome green, 8-bit
--          gdtri_h[15..08].in     = rwdin_h[31..24].q;
--          gdtri_h[07..00].in     = rwdin_h[23..16].q;
            !rdtrioe_l             = VCC;
            !gdtrioe_l             = rwwein_h.q # idle_h;
            !bdtrioe_l             = VCC;
```

```
            !rras_l             = rwallrasin_h.q;
            !rcasl_l            = rwallcasin_h.q;
            !rcash_l            = rwallcasin_h.q;
            !gras_l             = rwrasin_h.q # rwallrasin_h.q;
            !gcasl_l            = rwcasin_h.q # rwallcasin_h.q;
            !gcash_l            = rwcasin_h.q # rwallcasin_h.q;
            !bras_l             = rwallrasin_h.q;
            !bcasl_l            = rwallcasin_h.q;
            !bcash_l            = rwallcasin_h.q;
            enams_h             = VCC;
--          rwdout_h[31..24]    = gd_h[15..08];
--          rwdout_h[23..16]    = gd_h[07..00];
        WHEN rwmslsblu8 =>      -- data mode, LS monochrome blue, 8-bit
--          bdtri_h[15..08].in  = rwdin_h[15..08].q;
--          bdtri_h[07..00].in  = rwdin_h[07..00].q;
            !rdtrioe_l          = VCC;
            !gdtrioe_l          = VCC;
            !bdtrioe_l          = rwwein_h.q # idle_h;
            !rras_l             = rwallrasin_h.q;
            !rcasl_l            = rwallcasin_h.q;
            !rcash_l            = rwallcasin_h.q;
            !gras_l             = rwallrasin_h.q;
            !gcasl_l            = rwallcasin_h.q;
            !gcash_l            = rwallcasin_h.q;
            !bras_l             = rwrasin_h.q # rwallrasin_h.q;
            !bcasl_l            = rwcasin_h.q # rwallcasin_h.q;
            !bcash_l            = rwcasin_h.q # rwallcasin_h.q;
            enals_h             = VCC;
--          rwdout_h[15..08]    = bd_h[15..08];
--          rwdout_h[07..00]    = bd_h[07..00];
        WHEN rwmsmsblu8 =>      -- data mode, MS monochrome blue, 8-bit
--          bdtri_h[15..08].in  = rwdin_h[31..24].q;
--          bdtri_h[07..00].in  = rwdin_h[23..16].q;
            !rdtrioe_l          = VCC;
            !gdtrioe_l          = VCC;
            !bdtrioe_l          = rwwein_h.q # idle_h;
            !rras_l             = rwallrasin_h.q;
            !rcasl_l            = rwallcasin_h.q;
            !rcash_l            = rwallcasin_h.q;
            !gras_l             = rwallrasin_h.q;
            !gcasl_l            = rwallcasin_h.q;
            !gcash_l            = rwallcasin_h.q;
            !bras_l             = rwrasin_h.q # rwallrasin_h.q;
            !bcasl_l            = rwcasin_h.q # rwallcasin_h.q;
            !bcash_l            = rwcasin_h.q # rwallcasin_h.q;
            enams_h             = VCC;
--          rwdout_h[31..24]    = bd_h[15..08];
--          rwdout_h[23..16]    = bd_h[07..00];
        WHEN OTHERS =>
            !rdtrioe_l          = VCC;
            !gdtrioe_l          = VCC;
            !bdtrioe_l          = VCC;
            !rras_l             = rwallrasin_h.q;
            !rcasl_l            = rwallcasin_h.q;
            !rcash_l            = rwallcasin_h.q;
            !gras_l             = rwallrasin_h.q;
            !gcasl_l            = rwallcasin_h.q;
            !gcash_l            = rwallcasin_h.q;
            !bras_l             = rwallrasin_h.q;
            !bcasl_l            = rwallcasin_h.q;
            !bcash_l            = rwallcasin_h.q;
            enals_h             = GND;
            enams_h             = GND;
--          rwdout_h[31..24]    = GND;
```

```
--          rwdout_h[23..16]     = GND;
      END CASE;
   ELSE
      !rdtrioe_l       = VCC;
      !gdtrioe_l       = VCC;
      !bdtrioe_l       = VCC;
      a_h[]            = wrain_h[].q;
      !we_l            = wrwein_h.q;
      CASE wrmuxselin_h[] IS
         WHEN wrmsred =>            -- monochrome red
            rdtri_h[].in         = wrdin_h[].q;
            !rras_l              = wrrasin_h.q # wrallrasin_h.q;
            !rcasl_l             = wrcasin_h.q # wrallcasin_h.q;
            !rcash_l             = wrcasin_h.q # wrallcasin_h.q;
            !gras_l              = wrallrasin_h.q;
            !gcasl_l             = wrallcasin_h.q;
            !gcash_l             = wrallcasin_h.q;
            !bras_l              = wrallrasin_h.q;
            !bcasl_l             = wrallcasin_h.q;
            !bcash_l             = wrallcasin_h.q;
         WHEN wrmsgrn =>            -- monochrome green
            gdtri_h[].in         = wrdin_h[].q;
            !rras_l              = wrallrasin_h.q;
            !rcasl_l             = wrallcasin_h.q;
            !rcash_l             = wrallcasin_h.q;
            !gras_l              = wrrasin_h.q # wrallrasin_h.q;
            !gcasl_l             = wrcasin_h.q # wrallcasin_h.q;
            !gcash_l             = wrcasin_h.q # wrallcasin_h.q;
            !bras_l              = wrallrasin_h.q;
            !bcasl_l             = wrallcasin_h.q;
            !bcash_l             = wrallcasin_h.q;
         WHEN wrmsblu =>            -- monochrome blue
            bdtri_h[].in         = wrdin_h[].q;
            !rras_l              = wrallrasin_h.q;
            !rcasl_l             = wrallcasin_h.q;
            !rcash_l             = wrallcasin_h.q;
            !gras_l              = wrallrasin_h.q;
            !gcasl_l             = wrallcasin_h.q;
            !gcash_l             = wrallcasin_h.q;
            !bras_l              = wrrasin_h.q # wrallrasin_h.q;
            !bcasl_l             = wrcasin_h.q # wrallcasin_h.q;
            !bcash_l             = wrcasin_h.q # wrallcasin_h.q;
         WHEN wrmsrsrvd =>          -- reserved
            !rras_l              = wrallrasin_h.q;
            !rcasl_l             = wrallcasin_h.q;
            !rcash_l             = wrallcasin_h.q;
            !gras_l              = wrallrasin_h.q;
            !gcasl_l             = wrallcasin_h.q;
            !gcash_l             = wrallcasin_h.q;
            !bras_l              = wrallrasin_h.q;
            !bcasl_l             = wrallcasin_h.q;
            !bcash_l             = wrallcasin_h.q;
         WHEN wrmslsredgrn =>       -- RGB LS red/grn
            rdtri_h[07..00].in   = wrdin_h[07..00].q;
            gdtri_h[07..00].in   = wrdin_h[15..08].q;
            !rras_l              = wrrasin_h.q # wrallrasin_h.q;
            !rcasl_l             = wrcasin_h.q # wrallcasin_h.q;
            !rcash_l             = wrallcasin_h.q;
            !gras_l              = wrrasin_h.q # wrallrasin_h.q;
            !gcasl_l             = wrcasin_h.q # wrallcasin_h.q;
            !gcash_l             = wrallcasin_h.q;
            !bras_l              = wrallrasin_h.q;
            !bcasl_l             = wrallcasin_h.q;
            !bcash_l             = wrallcasin_h.q;
```

```
        WHEN wrmslsblu =>          -- RGB LS blue
            bdtri_h[07..00].in     = wrdin_h[07..00].q;
            !rras_l                = wrallrasin_h.q;
            !rcasl_l               = wrallcasin_h.q;
            !rcash_l               = wrallcasin_h.q;
            !gras_l                = wrallrasin_h.q;
            !gcasl_l               = wrallcasin_h.q;
            !gcash_l               = wrallcasin_h.q;
            !bras_l                = wrrasin_h.q # wrallrasin_h.q;
            !bcasl_l               = wrcasin_h.q # wrallcasin_h.q;
            !bcash_l               = wrallcasin_h.q;
        WHEN wrmsmsredgrn =>       -- RGB MS red/grn
            rdtri_h[15..08].in     = wrdin_h[07..00].q;
            gdtri_h[15..08].in     = wrdin_h[15..08].q;
            !rras_l                = wrrasin_h.q # wrallrasin_h.q;
            !rcasl_l               = wrallcasin_h.q;
            !rcash_l               = wrcasin_h.q # wrallcasin_h.q;
            !gras_l                = wrrasin_h.q # wrallrasin_h.q;
            !gcasl_l               = wrallcasin_h.q;
            !gcash_l               = wrcasin_h.q # wrallcasin_h.q;
            !bras_l                = wrallrasin_h.q;
            !bcasl_l               = wrallcasin_h.q;
            !bcash_l               = wrallcasin_h.q;
        WHEN wrmsmsblu =>          -- RGB MS blue
            bdtri_h[15..08].in     = wrdin_h[07..00].q;
            !rras_l                = wrallrasin_h.q;
            !rcasl_l               = wrallcasin_h.q;
            !rcash_l               = wrallcasin_h.q;
            !gras_l                = wrallrasin_h.q;
            !gcasl_l               = wrallcasin_h.q;
            !gcash_l               = wrallcasin_h.q;
            !bras_l                = wrrasin_h.q # wrallrasin_h.q;
            !bcasl_l               = wrallcasin_h.q;
            !bcash_l               = wrcasin_h.q # wrallcasin_h.q;
        WHEN OTHERS =>
            rdtri_h[15..08].in     = GND;
            gdtri_h[15..08].in     = GND;
            bdtri_h[15..08].in     = GND;
            !rras_l                = wrallrasin_h.q;
            !rcasl_l               = wrallcasin_h.q;
            !rcash_l               = wrallcasin_h.q;
            !gras_l                = wrallrasin_h.q;
            !gcasl_l               = wrallcasin_h.q;
            !gcash_l               = wrallcasin_h.q;
            !bras_l                = wrallrasin_h.q;
            !bcasl_l               = wrallcasin_h.q;
            !bcash_l               = wrallcasin_h.q;
    END CASE;
END IF;

%=========================================================================
Busy Counter

In order to insure that the DRAM's data lines are in a valid state
at all times, we maintain a 'busy' counter. It activates whenever
a DRAM cycle is detected and deactivates sometime after the cycle
ends.
=========================================================================%
busycnt_h[].clk     = bpclk_r;          -- non-global clock since 'rwrasin_h.q' is non-global
!busycnt_h[].prn    = !clr_l;
IF rwrasin_h.q THEN                     -- reset to zero
    busycnt_h[].d = GND;
ELSIF !idle_h THEN                      -- increment
    busycnt_h[].d = busycnt_h[].q + 1;
```

```
    ELSE                                        -- hold last value
        busycnt_h[].d = busycnt_h[].q;
    END IF;
    idle_h  = (busycnt_h[] == B"11");           -- LCELL

END; % SMUX.TDF %
```

```
%=============================================================================
WRITE.TDF - Frame Capture Board Write Controller
Copyright (c) 1996, Kensal Corporation

Revision History:
1.00    K.W. Crocker    11 Jun 96
    Initial writing.
1.01    K.W. Crocker    12 Jun 96
    Remove address mux (it belongs in Super Mux chip).
2.00    K.W. Crocker    22 Aug 96
    Done. Pins anchored. PCB released to fab
=============================================================================%
TITLE "Frame Capture Board Write Controller";

-- Values for ptnum_h[1..0]
CONSTANT rwCtrl          = B"00";    -- PCI & Read/Write Controller (ie. this chip)
CONSTANT wrCtrl          = B"01";    -- Write Controller

-- Longword offset within passthru region
CONSTANT wrCtrlReg       = H"0";     -- 0x00 >> 2 = 0x0
CONSTANT wrStatReg       = H"1";     -- 0x04 >> 2 = 0x1
CONSTANT wrPixReg        = H"2";     -- 0x08 >> 2 = 0x2
CONSTANT wrPixCnt        = H"3";     -- 0x0C >> 2 = 0x3 (read only)
CONSTANT wrAdrInc        = H"4";     -- 0x10 >> 2 = 0x4
CONSTANT wrAdrCnt        = H"5";     -- 0x14 >> 2 = 0x5

-- adr_h[6..2] Constants
CONSTANT APTD            = B"01011"; -- 0x2C >> 2 = 0x0B

-- Values for mode_h[1..0]
CONSTANT monored         = B"00";    -- monochrome red
CONSTANT monogreen       = B"01";    -- monochrome green
CONSTANT monoblue        = B"10";    -- monochrome blue
CONSTANT rgbmode         = B"11";    -- tri-color RGB time-multiplexed

-- Values for wrmuxsel_h[2..0] (Super Mux Control Signals)
CONSTANT msred           = B"000";   -- monochrome red
CONSTANT msgreen         = B"001";   -- monochrome green
CONSTANT msblue          = B"010";   -- monochrome blue
CONSTANT msrsrvd         = B"011";   -- reserved
CONSTANT mslsredgreen    = B"100";   -- RGB LS red/grn
CONSTANT mslsblue        = B"101";   -- RGB LS blue
CONSTANT msmsredgreen    = B"110";   -- RGB MS red/grn
CONSTANT msmsblue        = B"111";   -- RGB MS blue

SUBDESIGN write (
    -- Clock and Asynchronous Reset Inputs
    36mhz_r              : INPUT;        -- 36 MHz crystal input (global)
    bpclk_r              : INPUT;        -- 33 MHz buffered PCI clock (global)
    33mhz_r              : INPUT;        -- 33 MHz crystal (Hotlink Tx clk)
    ckr_r                : INPUT;        -- 33 Mhz PLL (Hotlink Rx clock)
    gclr_l               : INPUT;        -- clear from PCI controller (global)

    -- AMCC S5933 Passthru Interface Signals
    ptatn_l              : INPUT;        -- pass thru cycle input
    ptburst_l            : INPUT;        -- burst access input
    ptnum_h[1..0]        : INPUT;        -- base address register number
    ptbe_l[3..0]         : INPUT;        -- requested byte enables
    ptwr_h               : INPUT;        -- write (ptrd_l) input
    ptadr_l              : BIDIR;        -- force read of address reg (NOT slow slew rate)
    ptrdy_l              : BIDIR;        -- ready output (NOT slow slew rate)

    -- AMCC S5933 Add-On Bus Interface Signals
    dq_h[31..00]         : BIDIR;        -- data bus (slow slew rate)
```

```
    adr_h[6..2]          : BIDIR;        -- register address (NOT slow slew rate)
    select_l             : BIDIR;        -- cycle start (NOT slow slew rate)
    wr_l                 : BIDIR;        -- write strobe (NOT slow slew rate)
    rd_l                 : BIDIR;        -- read strobe (NOT slow slew rate)
    busgrnt_l            : BIDIR;        -- dq_h[] bus grant

    -- Interrupt to PCI Controller
    wrintout_h           : OUTPUT;       -- Write Controller Interrupt (slow slew rate)

    -- Transmit Hotlink Signals
    td_h[7..0]           : OUTPUT;       -- (slow slew rate)
    tsc_h                : OUTPUT;       -- high -> command, low -> data (slow slew rate)
    tsvs_h               : OUTPUT;       -- send violation symbol (slow slew rate)
    tbisten_l            : OUTPUT;       -- BIST enable (slow slew rate)
    tena_l               : OUTPUT;       -- enable (used to write data) (slow slew rate)
    tenn_l               : OUTPUT;       -- enable next (used for BIST) (slow slew rate)
    trp_l                : INPUT;        -- read pulse (currently unused)

    -- Receive Hotlink Signals
    rcd_h                : INPUT;        -- receive carrier detected
    rrvs_h               : INPUT;        -- received violation symbol
    rbisten_l            : OUTPUT;       -- BIST enable (slow slew rate)
    rselb_l              : OUTPUT;       -- select "B" input (slow slew rate)
    rrf_h                : OUTPUT;       -- reframe enable (slow slew rate)
    rrdy_l               : INPUT;        -- ready pulse

    -- Receive FIFO Common Signals
    rfrst_l              : OUTPUT;       -- (slow slew rate)

    -- Receive FIFO Signals (write side)
    enrfwen_l            : OUTPUT;       -- enables rfwen_l if rrdy_l is active.
    rfwen_l              : INPUT;
    rfir_l               : INPUT;

    -- Receive FIFO Signals (read side)
    enrfren_l            : OUTPUT;       -- enables rfren_l if rfor_l is active.
    rfsc_h               : INPUT;        -- read FIFO select cmd/~data
    rfd_h[7..0]          : INPUT;        -- read FIFO data [7..0]
    rfren_l              : INPUT;
    rfor_l               : INPUT;

    -- Interface to Super Mux
    writebank1_h         : OUTPUT;       -- H -> write controller hooked up to DRAM #1 (slow slew rate)
    wrmuxsel_h[2..0]     : OUTPUT;       -- (slow slew rate)
    wrd_h[15..00]        : OUTPUT;       -- (slow slew rate)
    wra_h[9..0]          : OUTPUT;
    wrras_h              : OUTPUT;       -- (slow slew rate)
    wrcas_h              : OUTPUT;       -- (slow slew rate)
    wrallras_h           : OUTPUT;       -- (slow slew rate)
    wrallcas_h           : OUTPUT;       -- (slow slew rate)
    wrwe_h               : OUTPUT;       -- (slow slew rate)

    -- Diagnostic Signals
    rbusy_l              : OUTPUT;       -- LED status output (Green)
    tbusy_l              : OUTPUT;       -- LED status output (Green)
    spare_l[1..0]        : OUTPUT;
    rfshforce_h          : INPUT;        -- force refresh
)

VARIABLE
    -- Global Signals
    clk36_r              : NODE;
    clk_r                : NODE;
    clr_l                : NODE;
```

```
-- AMCC S5933 Passthru Interface Signals
ptadrtri_l          : TRI;
ptrdytri_l          : TRI;
ptatn_h             : LCELL;
ptburst_h           : LCELL;
ptbe_h[3..0]        : LCELL;
ptnumd_h[1..0]      : LCELL;

-- AMCC S5933 Add-On Bus Interface Signals
dqoutff_h[31..00]   : DFFE;
dqtri_h[31..00]     : TRI;
dqoe_l              : LCELL;
adrtri_h[6..2]      : TRI;
selecttri_l         : TRI;
wrtri_l             : TRI;
rdtri_l             : TRI;
busgrnttri_l        : TRI;

-- Passthru Address Register
ptaddr_h[3..0]      : DFF;      -- longword offset within passthru region
ptadrinc_h          : NODE;

-- Transmit Hotlink Signals
twr_h               : DFF;
td_h[7..0]          : DFF;
tsvs_h              : DFF;
tsc_h               : DFF;
tena_l              : DFF;
tenn_l              : DFF;
trpdet_h            : DFF;
trplat_h            : SRFF;
tbisten_l           : DFF;

-- Receive Hotlink Signals
rvs_h               : SRFF;     -- receive violation symbol detected
rrf_h               : DFF;
rselb_l             : DFF;
rbisten_l           : DFF;

-- Receive FIFO Signals (write side)
enrfwen_l           : DFF;
rfov_h              : SRFF;     -- receive FIFO overflow status bit

-- LED Status Output Signals
rbusycnt_h[3..0]    : DFF;
tbusycnt_h[3..0]    : DFF;

-- Receive FIFO Signals (read side)
rfrd_h              : DFF;      -- Receive FIFO Read Word Command Bit
--  rfscin_h            : DFFE;
--  rfdin_h[7..0]       : DFFE;
rfsca_h             : DFFE;
rfda_h[7..0]        : DFFE;
rfdb_h[7..0]        : DFFE;
rfdc_h[15..00]      : DFFE;
setoddbyte_h        : NODE;
oddbyte_h           : SRFF;     -- Odd number of bytes received
setwrint_h          : NODE;
wrint_h             : SRFF;     -- Write interrupt

-- RAS/CAS Multiplexors
rgbsel_h[1..0]      : DFF;
```

```
-- Super Mux Signals
writebank1_h         : DFF;

-- Register Select Signals
clkce1_h             : DFF;      -- edge detectors
clkce2_h             : DFF;
clkce_h              : NODE;
clk36ce1_h           : DFF;
clk36ce2_h           : DFF;
clk36ce_h            : NODE;
clkrce1_h            : DFF;
clkrce2_h            : DFF;
ckrce_h              : NODE;
clk33ce1_h           : DFF;
clk33ce2_h           : DFF;
clk33ce_h            : NODE;

ctrlregclkwe_h       : LCELL;    -- ctrl reg write enables
ctrlregclk36we_h     : LCELL;
ctrlregckrwe_h       : LCELL;
ctrlregclk33we_h     : LCELL;
wrpixregwe_h         : LCELL;    -- wrpixreg_h[] write enable

-- Control Register Signals
spare_l[1..0]        : DFF;
wrinten_h            : DFF;
xfren_h              : DFF;
mode_h[1..0]         : DFF;

-- Refresh Counter
rfshcnt_h[8..0]      : DFF;      -- Refresh counter
rfshreq_h            : SRFF;

-- Write Pixel Register
wrpixreg_h[12..00]   : DFF;      -- Write pixel register (DRAM writes/video line)

-- Write Pixel Counter
wrpixcnt_h[12..00]   : DFF;      -- Write pixel counter
wrpixcnttc_h         : LCELL;
endline_h            : DFF;      -- was NODE

-- Write Address Increment
wradrincwe_h         : LCELL;    -- wradrinc_h[] write enable
wradrinc_h[12..00]   : DFF;      -- Write address increment

-- Write Address Counter
wradrcntwe_h         : LCELL;    -- wradrcnt_h[] write enable
wradrcnt_h[19..00]   : DFF;      -- Write address counter
wradrcnttc_h         : LCELL;       -- Term cnt on [09..00]
endrow_h             : DFF;

-- Summation Register
sum_h[19..00]        : LCELL;    -- Sum of write adr cntr and write adr incr
carry_h              : LCELL;

-- Pass-Through State Machine
ptSM                 : MACHINE OF BITS (
                         busgrnt_h, select_h, rd_h, wr_h, ptadr_h, ptrdy_h, dqoe_h, tsoe_l
                       ) WITH STATES (
                         pt00    = B"00000001",
                         pt01    = B"11001000",
                         pt02    = B"11001000",
                         pt03    = B"11100000",
                         pt04    = B"11100000",
```

```
                  pt05     = B"11100100",
                  pt06     = B"11010010",
                  pt07     = B"11010010",
                  pt08     = B"11010110",
                  pt09     = B"00000000"
              );

-- Read FIFO Pipeline State Machine
startcycle_h          : NODE;      -- was LCELL
willgo_h              : NODE;      -- was LCELL
cdavail_h             : NODE;
endxfr_h              : NODE;
qendxfr_h             : SRFF;
pSM                   : MACHINE OF BITS (
                          enrfren_h, cclken_h, cfull_h, bfull_h, afull_h
                      ) WITH STATES (
                          p00          = B"00000",
                          p01          = B"10000",
                          p02          = B"10001",
                          p03          = B"11011",
                          p04          = B"10101",
                          p05          = B"10100",
                          p06          = B"00011",
                          p07          = B"00001",
                          p08          = B"00101",
                          p09          = B"00100",
                          p10          = B"10000",
                          p11          = B"00000"
                      );

-- CBR State Machine
cbrSM                 : MACHINE OF BITS (
                          wrallcas_h, wrallras_h, cbrdone_h
                      ) WITH STATES (
                          cIdle        = B"000",
                          cCBR1        = B"100",
                          cCBR2        = B"110",
                          cCBR3        = B"010",
                          cCBR4        = B"011"
                      );

-- DRAM State Machine
canras_h              : LCELL;
willcas_h             : LCELL;
stop_h                : LCELL;         -- indicates need to close row.
pc_count_h            : NODE;
docount_h             : DFF;
count_h               : DFF;
load_h                : DFF;
dramSM                : MACHINE OF BITS (
                          wrras_h, wrcas_h, colsel_h,
                          wrwe_h, add_h, docbr_h
                      ) WITH STATES (
                          dIdle1       = B"000000",
                          dRfsh1       = B"000001",

                          dRas         = B"100000",
                          dColSel      = B"101100",
                          dCas         = B"111100",

                          dNewLn       = B"000000",
                          dWait        = B"000000",
                          dAdd         = B"000010",
```

```
                              dRasPre      = B"000000",
                              dIdle2       = B"000000",
                              dRfsh2       = B"000001"
                          );

BEGIN
    -- Global Signals
    clk36_r              = GLOBAL (36mhz_r);     -- 36 MHz crystal oscillator
    clk_r                = GLOBAL (bpclk_r);     -- 33 MHz buffered PCI clock
    clr_l                = GLOBAL (gclr_l);

    %================================================================
    Pass Thru Transactions
    ================================================================%

    -- Passthru Address Register
    ptaddr_h[].clk       = clk_r;
    !ptaddr_h[].clrn     = !clr_l;
    IF pt02 THEN
        ptaddr_h[].d = dq_h[05..02];
    ELSIF ptadrinc_h THEN
        ptaddr_h[].d = ptaddr_h[].q + 1;
    ELSE
        ptaddr_h[].d = ptaddr_h[].q;
    END IF;
    ptadrinc_h = ptrdy_h & ptatn_h;

    -- Input Bits
    ptatn_h       = !ptatn_l;      -- (LCELL) compensate for clk buf and 0 hld
    ptburst_h     = !ptburst_l;    -- (LCELL) compensate for clk buf and 0 hld
    ptbe_h[]      = !ptbe_l[];     -- (LCELL) compensate for clk buf and 0 hld
    ptnumd_h[]    = ptnum_h[];     -- (LCELL) compensate for clk buf and 0 hld

    -- Output Bits
    adrtri_h[].in        = APTD;
    adrtri_h[].oe        = !tsoe_l;
    adr_h[]              = adrtri_h[].out;

    !selecttri_l.in      = select_h;
    selecttri_l.oe       = !tsoe_l;
    select_l             = selecttri_l.out;

    !rdtri_l.in          = rd_h;
    rdtri_l.oe           = !tsoe_l;
    rd_l                 = rdtri_l.out;

    !wrtri_l.in          = wr_h;
    wrtri_l.oe           = !tsoe_l;
    wr_l                 = wrtri_l.out;

    !ptadrtri_l.in       = ptadr_h;
    ptadrtri_l.oe        = !tsoe_l;
    ptadr_l              = ptadrtri_l.out;

    !ptrdytri_l.in       = ptrdy_h;
    ptrdytri_l.oe        = !tsoe_l;
    ptrdy_l              = ptrdytri_l.out;

    !busgrnttri_l.in     = busgrnt_h;
    busgrnttri_l.oe      = !tsoe_l;
    busgrnt_l            = busgrnttri_l.out;

    %================================================================
    Pass Thru State Machine
    ================================================================%
```

```
ptSM.clk      = clk_r;
ptSM.reset    = !clr_l;

-- The LCELL delay here is actually beneficial. It helps prevent dq_h[]
-- bus conflict during the time 'ptadr_h' is deactivating and the
-- write controller begins driving the dq_h[] bus with data.
dqoe_l        = !dqoe_h;  -- LCELL

CASE ptSM IS
    WHEN pt00 =>          -- <nothing> active
        IF busgrnt_l & ptatn_h &
            (ptnumd_h[] == wrCtrl) & (ptbe_h[] == B"1111") THEN ptSM = pt01;
        END IF;

    -- Get Passthru Address
    WHEN pt01 =>          -- busgrnt_h, select_h, ptadr_h, tsoe_l active
        ptSM = pt02;
    WHEN pt02 =>          -- busgrnt_h, select_h, ptadr_h, tsoe_l active
        IF ptwr_h THEN ptSM = pt03;
        ELSE ptSM = pt06;
        END IF;

    -- Passthru Write Operations
    WHEN pt03 =>          -- busgrnt_h, select_h, rd_h, tsoe_l active
        IF !ptburst_h & !ptatn_h THEN ptSM = pt09;
        ELSE ptSM = pt04;
        END IF;
    WHEN pt04 =>          -- busgrnt_h, select_h, rd_h, tsoe_l active
        ptSM = pt05;
    WHEN pt05 =>          -- busgrnt_h, select_h, rd_h, ptrdy_h, tsoe_l active
        IF ptatn_h THEN ptSM = pt03;
        END IF;

    -- Passthru Read Operations
    WHEN pt06 =>          -- busgrnt_h, select_h, wr_h, dqoe_h, tsoe_l active
        IF !ptburst_h & !ptatn_h THEN ptSM = pt09;
        ELSE ptSM = pt07;
        END IF;
    WHEN pt07 =>          -- busgrnt_h, select_h, wr_h, dqoe_h, tsoe_l active
        ptSM = pt08;
    WHEN pt08 =>          -- busgrnt_h, select_h, wr_h, ptrdy_h, dqoe_h, tsoe_l active
        IF ptatn_h THEN ptSM = pt06;
        END IF;

    -- Drive Control Signals Inactive Before High-Z
    WHEN pt09 =>          -- tsoe_l active
        ptSM = pt00;
END CASE;

-- Local Bus Interface
dqtri_h[].in        = dqoutff_h[].q;
dqtri_h[].oe        = !dqoe_l;
dq_h[]              = dqtri_h[].out;

%=====================================================================
Multiple Clock Synchronization Logic
=====================================================================%
-- Clock Enable Edge Detector for bits synch'd by buffered 33MHz PCI clock
clkce1_h.clk        = clk_r;
!clkce1_h.clrn      = !clr_l;
clkce1_h.d          = rd_h # wr_h;

clkce2_h.clk        = clk_r;
!clkce2_h.clrn      = !clr_l;
```

```
clkce2_h.d          = clkce1_h.q;

clkce_h             = clkce1_h.q & !clkce2_h.q;

-- Clock Enable Edge Detector for bits synch'd by 36 MHz crystal oscillator
clk36ce1_h.clk      = clk36_r;
!clk36ce1_h.clrn    = !clr_l;
clk36ce1_h.d        = rd_h # wr_h;

clk36ce2_h.clk      = clk36_r;
!clk36ce2_h.clrn    = !clr_l;
clk36ce2_h.d        = clk36ce1_h.q;

clk36ce_h           = clk36ce1_h.q & !clk36ce2_h.q;

-- Write Enable Edge Detector for bits synch'd by 33 Mhz PLL clock from Rx Hotlink
clkrce1_h.clk       = ckr_r;
!clkrce1_h.clrn     = !clr_l;
clkrce1_h.d         = rd_h # wr_h;

clkrce2_h.clk       = ckr_r;
!clkrce2_h.clrn     = !clr_l;
clkrce2_h.d         = clkrce1_h.q;

ckrce_h             = clkrce1_h.q & !clkrce2_h.q;

-- Write Enable Edge Detector for bits synch'd by 33 Mhz clock from onboard oscillator
clk33ce1_h.clk      = 33mhz_r;
!clk33ce1_h.clrn    = !clr_l;
clk33ce1_h.d        = rd_h # wr_h;

clk33ce2_h.clk      = 33mhz_r;
!clk33ce2_h.clrn    = !clr_l;
clk33ce2_h.d        = clk33ce1_h.q;

clk33ce_h           = clk33ce1_h.q & !clk33ce2_h.q;

%=====================================================================
Register Read Logic
=====================================================================%
-- Receive FIFO Overflow Status Bit
rfov_h.clk      = ckr_r;        -- 33 MHz PLL clock from Rx Hotlink
!rfov_h.clrn    = !clr_l;
rfov_h.r        = rfov_h.q & !dq_h[02] & ckrce_h & rd_h & (ptaddr_h[] == wrStatReg)
                & !(rfir_l & !rfwen_l);
rfov_h.s        = rfir_l & !rfwen_l;

-- Receive FIFO Violation Symbol Received Status Bit
rvs_h.clk       = ckr_r;        -- 33 MHz PLL clock from Rx Hotlink
!rvs_h.clrn     = !clr_l;
rvs_h.r         = rvs_h.q & !dq_h[03] & ckrce_h & rd_h & (ptaddr_h[] == wrStatReg)
                & !(rrvs_h & !rrdy_l);
rvs_h.s         = rrvs_h & !rrdy_l;

-- Transmit Hotlink Read Pulse Detector
trpdet_h.clk    = trp_l;
trpdet_h.d      = VCC;
!trpdet_h.clrn  = !gclr_l # trplat_h.q;

-- Transmit Hotlink Read Pulse Status Bit
trplat_h.clk    = clk_r;        -- This can be any clock, since the read pulse is latched!
!trplat_h.clrn  = !clr_l;
trplat_h.r      = trplat_h.q & !dq_h[05] & clkce_h & rd_h & (ptaddr_h[] == wrStatReg);
trplat_h.s      = trpdet_h.q;
```

```
-- Odd Number of Bytes Transferred Status Bit
oddbyte_h.clk   = clk36_r;
!oddbyte_h.clrn = !clr_l;
oddbyte_h.r     = oddbyte_h.q & !dq_h[06] & clk36ce_h & rd_h & (ptaddr_h[] == wrStatReg)
                & !setoddbyte_h;
oddbyte_h.s     = setoddbyte_h;

-- Interrupt Flag
wrint_h.clk     = clk36_r;
!wrint_h.clrn   = !clr_l;
wrint_h.r       = wrint_h.q & !dq_h[07] & clk36ce_h & rd_h & (ptaddr_h[] == wrStatReg)
                & !setwrint_h;
wrint_h.s       = setwrint_h;

-- Interrupt Output Signal
wrintout_h      = wrint_h.q & wrinten_h.q;

-- Register Read Logic
dqoutff_h[].clk     = clk_r;
!dqoutff_h[].clrn   = !clr_l;
dqoutff_h[].ena     = LCELL (clkce_h & wr_h);
CASE ptaddr_h[] IS
    WHEN wrCtrlReg =>
        dqoutff_h[31..27].d = GND;
        dqoutff_h[26..25].d = !spare_l[].q;
        dqoutff_h[24].d     = tsc_h.q;
        dqoutff_h[23..16].d = td_h[].q;
        dqoutff_h[14].d     = GND;          -- twr_h always returns '0'
        dqoutff_h[13].d     = rfrd_h.q;
        dqoutff_h[12].d     = writebank1_h;
        dqoutff_h[11].d     = wrinten_h.q;
        dqoutff_h[10].d     = xfren_h.q;
        dqoutff_h[09..08].d = mode_h[].q;
        dqoutff_h[07].d     = rrf_h.q;
        dqoutff_h[06].d     = !rselb_l.q;
        dqoutff_h[05].d     = tsvs_h.q;
        dqoutff_h[04].d     = !tenn_l.q;
        dqoutff_h[03].d     = !rbisten_l.q;
        dqoutff_h[02].d     = !tbisten_l.q;
        dqoutff_h[01].d     = GND;          -- rfrst_h always returns '0'
        dqoutff_h[00].d     = !enrfwen_l.q;
    WHEN wrStatReg =>
        dqoutff_h[31..25].d = GND;
        dqoutff_h[24].d     = rfsca_h.q;
        dqoutff_h[23..16].d = rfda_h[7..0].q;
        dqoutff_h[15..08].d = GND;
        dqoutff_h[07].d     = wrint_h.q;
        dqoutff_h[06].d     = oddbyte_h.q;
        dqoutff_h[05].d     = trplat_h.q;
        dqoutff_h[04].d     = rcd_h;
        dqoutff_h[03].d     = rvs_h.q;
        dqoutff_h[02].d     = rfov_h.q;
        dqoutff_h[01].d     = !rfor_l;
        dqoutff_h[00].d     = !rfir_l;
    WHEN wrPixReg =>
        dqoutff_h[31..13].d = GND;
        dqoutff_h[12..00].d = wrpixreg_h[].q;
    WHEN wrPixCnt =>
        dqoutff_h[31..13].d = GND;
        dqoutff_h[12..00].d = wrpixcnt_h[].q;
    WHEN wrAdrInc =>
        dqoutff_h[31..13].d = GND;
        dqoutff_h[12..00].d = wradrinc_h[].q;
```

```
    WHEN wrAdrCnt =>
        dqoutff_h[31..20].d = GND;
        dqoutff_h[19..00].d = wradrcnt_h[].q;
END CASE;

%=======================================================================
Register Write Logic
=======================================================================%
--------------------------------------------------------------------------
-- First, handle control register bits synchronized with 'clk_r'
-- This comes from the PCI bus. It is nominally 33.333Mhz.
--------------------------------------------------------------------------
ctrlregclkwe_h  = clkce_h & rd_h & (ptaddr_h[] == wrCtrlReg);    -- LCELL

spare_l[].clk        = clk_r;
!spare_l[].prn       = !clr_l;
writebank1_h.clk     = clk_r;
!writebank1_h.clrn   = !clr_l;
wrinten_h.clk        = clk_r;
!wrinten_h.clrn      = !clr_l;

CASE ctrlregclkwe_h IS
    WHEN B"0" =>
        !spare_l[].d     = !spare_l[].q;
        writebank1_h.d   = writebank1_h.q;
        wrinten_h.d      = wrinten_h.q;
    WHEN B"1" =>
        !spare_l[].d     = dq_h[26..25];
        writebank1_h.d   = dq_h[12];
        wrinten_h.d      = dq_h[11];
END CASE;

--------------------------------------------------------------------------
-- Next, handle control register bits synchronized with 'clk36_r'
-- This comes from a 36MHz onboard oscillator, which is used to clock
-- the output side of the receive FIFO.
--------------------------------------------------------------------------
ctrlregclk36we_h    = clk36ce_h & rd_h & (ptaddr_h[] == wrCtrlReg); -- LCELL

-- Receive FIFO Read Word Command Bit
rfrd_h.clk      = clk36_r;
!rfrd_h.clrn    = !clr_l;
IF (pSM == p10) & cdavail_h THEN
    rfrd_h.d    = GND;
ELSIF ctrlregclk36we_h THEN
    rfrd_h.d    = dq_h[13];
ELSE
    rfrd_h.d    = rfrd_h.q;
END IF;

-- Transfer Enable Command Bit
xfren_h.clk     = clk36_r;
!xfren_h.clrn   = !clr_l;
IF xfren_h.q & endxfr_h THEN    -- transfer ended!
    xfren_h.d   = GND;
    setwrint_h  = VCC;
ELSIF ctrlregclk36we_h THEN
    xfren_h.d   = dq_h[10];
    setwrint_h  = xfren_h.q & !dq_h[10];
ELSE
    xfren_h.d   = xfren_h.q;
END IF;

mode_h[].clk    = clk36_r;
```

```
!mode_h[].clrn  = !clr_l;

CASE ctrlregclk36we_h IS
    WHEN B"0" =>
        -- rfrd_h handled separately above
        -- xfren_h handled separately above
        mode_h[].d       = mode_h[].q;
        !rfrst_l         = GND;
    WHEN B"1" =>
        -- rfrd_h handled separately above
        -- xfren_h handled separately above
        mode_h[].d       = dq_h[09..08];
        !rfrst_l         = dq_h[01];
END CASE;

------------------------------------------------------------------------
-- Next, handle control register bits synchronized with 'ckr_r'
-- This comes from the read Hotlink's PLL and is used to clock the
-- input side of the receive FIFO.
------------------------------------------------------------------------
ctrlregckrwe_h  = ckrce_h & rd_h & (ptaddr_h[] == wrCtrlReg);   -- LCELL

rrf_h.clk       = ckr_r;        -- 33 Mhz PLL clock from Rx Hotlink
!rrf_h.clrn     = !clr_l;
rselb_l.clk     = ckr_r;        -- 33 Mhz PLL clock from Rx Hotlink
!rselb_l.prn    = !clr_l;
rbisten_l.clk   = ckr_r;        -- 33 Mhz PLL clock from Rx Hotlink
!rbisten_l.prn  = !clr_l;
enrfwen_l.clk   = ckr_r;        -- 33 Mhz PLL clock from Rx Hotlink
!enrfwen_l.prn  = !clr_l;

CASE ctrlregckrwe_h IS
    WHEN B"0" =>
        rrf_h.d          = rrf_h.q;
        !rselb_l.d       = !rselb_l.q;
        !rbisten_l.d     = !rbisten_l.q;
        !enrfwen_l.d     = !enrfwen_l.q;
    WHEN B"1" =>
        rrf_h.d          = dq_h[07];
        !rselb_l.d       = dq_h[06];
        !rbisten_l.d     = dq_h[03];
        !enrfwen_l.d     = dq_h[00];
END CASE;

------------------------------------------------------------------------
-- Lastly, handle control register bits synchronized with '33mhz_r'
-- This comes from a 33.333MHz onboard oscillator. We probably could have
-- gotten by using the 'clk_r' signal from the PCI bus. Using our own
-- oscillator means we can specify an oscillator with spec'ed phase noise
-- if needed. This signal clocks the transmit Hotlink.
------------------------------------------------------------------------
ctrlregclk33we_h    = clk33ce_h & rd_h & (ptaddr_h[] == wrCtrlReg); -- LCELL

twr_h.clk       = 33mhz_r;
!twr_h.clrn     = !clr_l;
tsc_h.clk       = 33mhz_r;
!tsc_h.clrn     = !clr_l;
td_h[].clk      = 33mhz_r;
!td_h[].clrn    = !clr_l;
tsvs_h.clk      = 33mhz_r;
!tsvs_h.clrn    = !clr_l;
tenn_l.clk      = 33mhz_r;
!tenn_l.prn     = !clr_l;
tbisten_l.clk   = 33mhz_r;
```

```
!tbisten_l.prn  = !clr_l;

CASE ctrlregclk33we_h IS
    WHEN B"0" =>
        tsc_h.d          = tsc_h.q;
        td_h[].d         = td_h[].q;
        twr_h.d          = GND;      -- active for one 33mhz_r cycle only!
        tsvs_h.d         = tsvs_h.q;
        !tenn_l.d        = !tenn_l.q;
        !tbisten_l.d     = !tbisten_l.q;
    WHEN B"1" =>
        tsc_h.d          = dq_h[24];
        td_h[].d         = dq_h[23..16];
        twr_h.d          = dq_h[14];
        tsvs_h.d         = dq_h[05];
        !tenn_l.d        = dq_h[04];
        !tbisten_l.d     = dq_h[02];
END CASE;

-- 'tena_l' Output Bit
-- We could have outputted an inverted form of 'twr_h' directly,
-- but we delay one '33mhz_r' period to allow additional data setup
-- and to absolutely avoid any problem with metastability.
tena_l.clk      = 33mhz_r;
!tena_l.prn     = !clr_l;
!tena_l.d       = twr_h.q;

%======================================================================
Write Pixel Register (wrpixreg_h[12..00])
======================================================================%
wrpixregwe_h        = clk36ce_h & rd_h & (ptaddr_h[] == wrPixReg);  -- LCELL
wrpixreg_h[].clk    = clk36_r;
!wrpixreg_h[].clrn  = !clr_l;
CASE wrpixregwe_h IS
    WHEN B"0" =>          -- hold last value
        wrpixreg_h[].d = wrpixreg_h[].q;
    WHEN B"1" =>          -- load register
        wrpixreg_h[].d = dq_h[12..00];
END CASE;

%======================================================================
Write Pixel Counter/Register (wrpixcnt_h[12..00])
======================================================================%
wrpixcnt_h[].clk    = clk36_r;
!wrpixcnt_h[].clrn  = !clr_l;
CASE (load_h, count_h) IS
    WHEN B"01" =>          -- decrement counter
        wrpixcnt_h[].d = wrpixcnt_h[].q - 1;
    WHEN B"10" =>          -- load register/counter
        wrpixcnt_h[].d = wrpixreg_h[].q;
    WHEN OTHERS =>         -- hold last value
        wrpixcnt_h[].d = wrpixcnt_h[].q;
END CASE;
wrpixcnttc_h    = (wrpixcnt_h[] == 0);  -- LCELL
-- Since 'endline_h' goes active the cycle 'count_h' is active,
-- it will only be active for one clock because the counter
-- will rollover to a non-terminal count.
endline_h.clk   = clk36_r;
!endline_h.clrn = !clr_l;
endline_h.d     = pc_count_h & wrpixcnttc_h;

%======================================================================
Write Address Increment Register (wradrinc_h[12..00])
======================================================================%
```

```
wradrincwe_h        = clk36ce_h & rd_h & (ptaddr_h[] == wrAdrInc);  -- LCELL
wradrinc_h[].clk    = clk36_r;
!wradrinc_h[].clrn  = !clr_l;
CASE wradrincwe_h IS
    WHEN B"0" =>         -- hold last value
        wradrinc_h[].d = wradrinc_h[].q;
    WHEN B"1" =>         -- load register
        wradrinc_h[].d = dq_h[12..00];
END CASE;

%==================================================================
Write Address Counter/Register (wradrcnt_h[19..00])
==================================================================%
wradrcntwe_h        = clk36ce_h & rd_h & (ptaddr_h[] == wrAdrCnt);  -- LCELL
wradrcnt_h[].clk    = clk36_r;
!wradrcnt_h[].clrn  = !clr_l;

-- Write Address Counter/Register (part 1)
CASE (wradrcntwe_h, add_h, count_h) IS
    WHEN B"001" =>       -- increment counter by 1
        wradrcnt_h[09..00].d = wradrcnt_h[09..00].q + 1;
    WHEN B"010" =>       -- increment register by sum_h[]
        wradrcnt_h[09..00].d = sum_h[09..00];
    WHEN B"100" =>       -- load register
        wradrcnt_h[09..00].d = dq_h[09..00];
    WHEN OTHERS =>       -- hold last value
        wradrcnt_h[09..00].d = wradrcnt_h[09..00].q;
END CASE;
wradrcnttc_h     = (wradrcnt_h[09..00] == H"3FF");   -- LCELL

-- Since 'endrow_h' goes active the cycle 'count_h' is active,
-- it will only be active for one clock because the counter
-- will rollover to a non-terminal count.
endrow_h.clk     = clk36_r;
!endrow_h.clrn   = !clr_l;
endrow_h.d       = pc_count_h & wradrcnttc_h;

-- Write Address Counter/Register (part 2)
CASE (wradrcntwe_h, add_h, endrow_h) IS
    WHEN B"001" =>       -- increment counter by 1
        wradrcnt_h[19..10].d = wradrcnt_h[19..10].q + 1;
    WHEN B"010" =>       -- increment register by sum_h[]
        wradrcnt_h[19..10].d = sum_h[19..10];
    WHEN B"100" =>       -- load register
        wradrcnt_h[19..10].d = dq_h[19..10];
    WHEN OTHERS =>       -- hold last value
        wradrcnt_h[19..10].d = wradrcnt_h[19..10].q;
END CASE;

-- Sum Register (Note: carry_h is declared as an LCELL)
(carry_h, sum_h[09..00])     = (B"0", wradrcnt_h[09..00].q) + (B"0", wradrinc_h[09..00].q);
sum_h[19..10]                = wradrcnt_h[19..10].q + (B"0000000", wradrinc_h[12..10].q)
                             + (B"000000000", carry_h);

%==================================================================
DRAM Refresh Counter

Refreshes will be scheduled every 1/2**9 clocks. The clock period is
27.78 ns (36 MHz), so this equates to 14.222 us/CBR. At that rate,
the entire 1,024 rows will be refreshed in 14.564 ms. The spec. for
the part is 16 ms. For the SRFF 'rfshreq_h', sets have priority over
resets, so if another refresh cycle is queued on the same cycle that
'cbrdone_h' goes active, another refresh will be requested.
==================================================================%
```

```
rfshcnt_h[].clk      = clk36_r;
!rfshcnt_h[].clrn    = !clr_l;
rfshcnt_h[].d        = rfshcnt_h[].q + 1;

rfshreq_h.clk        = clk36_r;
!rfshreq_h.clrn      = !clr_l;
rfshreq_h.s          = (rfshcnt_h[].q == 2) # rfshforce_h;
rfshreq_h.r          = cbrdone_h & !((rfshcnt_h[].q == 2) # rfshforce_h);

%=========================================================================
Read FIFO Pipeline State Machine
=========================================================================%
pSM.clk        = clk36_r;
pSM.reset      = !clr_l;

!enrfren_l     = enrfren_h;
setoddbyte_h   = ((pSM == p02) # (pSM == p04)) & cdavail_h & rfsc_h;

-- An active 'startcycle_h' signals the DRAM state machine to begin
-- a RAS/CAS write cycle.
startcycle_h   = ((pSM == p06) # (pSM == p08) # (pSM == p09)) & xfren_h;

-- An active 'willgo_h' means that data will emerge from the pipeline
-- and be available to write to DRAM in 2 clocks (ie. dCAS in 2 clocks).
-- This means that if the dramSM is currently in dCAS AND !endline_h
-- AND !rfshreq_h AND !endrow_h THEN a DRAM write will occur using
-- fast page mode. The complement to this signal in the DRAM SM is
-- 'stop_h', which indicates that a FPM write is NOT possible.
willgo_h       = (pSM == p03) & xfren_h;

-- Command/Data Available
-- Due to timing constraints, 'cdavail_h' cannot be an LCELL
cdavail_h    = !rfren_l & LCELL (!rfsc_h # (rfd_h[] != H"05"));

-- 'endxfr_h' is active at the end of a transfer (a command symbol is
-- received) after the data in the pipeline has been written to DRAM.
-- It is only active one cycle.
endxfr_h     = (pSM == p11);

-- 'qendxfr_h' becomes active when 'xfren_h' is active and a command symbol is
-- received. It deactivates on the same clock that 'xfren_h' becomes inactive
-- after the last data is written.
qendxfr_h.clk    = clk36_r;
!qendxfr_h.clrn = !clr_l;
qendxfr_h.s      = xfren_h & cdavail_h & rfsc_h;
qendxfr_h.r      = !xfren_h # endxfr_h;

CASE pSM IS
    WHEN p00 =>      -- <nothing> active
        IF rfrd_h THEN pSM = p10;
        ELSIF xfren_h THEN pSM = p01;
        END IF;
    WHEN p01 =>      -- enrfren_h active
        IF !xfren_h THEN pSM = p00;          -- abort transfer!
        ELSIF cdavail_h THEN
            IF rfsc_h THEN pSM = p11;        -- normal exit.
            ELSE pSM = p02;
            END IF;
        END IF;
    WHEN p02 =>      -- enrfren_h, afull_h active
        IF !xfren_h THEN pSM = p00;          -- abort transfer!
        ELSIF cdavail_h THEN
            IF rfsc_h THEN pSM = p11;        -- error! odd bytes
            ELSE pSM = p06;
```

```
            END IF;
        END IF;
    WHEN p03 =>       -- enrfren_h, cclken_h, bfull_h, afull_h active
        IF !xfren_h THEN pSM = p00;         -- abort transfer!
        ELSIF cdavail_h THEN
            IF rfsc_h THEN pSM = p09;       -- normal exit, part 1
            ELSIF stop_h THEN pSM = p08;
            ELSE pSM = p04;
            END IF;
        ELSIF stop_h THEN pSM = p09;
        ELSE pSM = p05;
        END IF;
    WHEN p04 =>       -- enrfren_h, cfull_h, afull_h active
        IF !xfren_h THEN pSM = p00;         -- abort transfer!
        ELSIF cdavail_h THEN
            IF rfsc_h THEN pSM = p11;       -- error! odd bytes
            ELSE pSM = p03;
            END IF;
        ELSE pSM = p07;
        END IF;
    WHEN p05 =>       -- enrfren_h, cfull_h active
        IF !xfren_h THEN pSM = p00;         -- abort transfer!
        ELSIF cdavail_h THEN
            IF rfsc_h THEN pSM = p11;       -- normal exit.
            ELSE pSM = p07;
            END IF;
        ELSE pSM = p01;
        END IF;
    WHEN p06 =>       -- bfull_h, afull_h active
        IF !xfren_h THEN pSM = p00;         -- abort transfer!
        ELSIF canras_h THEN pSM = p03;
        END IF;
    WHEN p07 =>       -- afull_h active
        IF !xfren_h THEN pSM = p00;
        ELSE pSM = p02;
        END IF;
    WHEN p08 =>       -- cfull_h, afull_h active
        IF !xfren_h THEN pSM = p00;         -- abort transfer!
        ELSIF willcas_h THEN pSM = p02;
        END IF;
    WHEN p09 =>       -- cfull_h active
        IF !xfren_h THEN pSM = p00;         -- abort transfer!
        ELSIF willcas_h THEN
            IF qendxfr_h THEN pSM = p11;    -- normat exit, part 2
            ELSE pSM = p01;
            END IF;
        END IF;
    WHEN p10 =>       -- enrfren_h active
        IF !rfrd_h # cdavail_h THEN pSM = p00;
        END IF;

    -- End of transfer detected (ie. cdavail_h & rfsc_h)
    -- May be a normal or oddbyte exit, but not aborted!
    WHEN p11 =>       -- <nothing> active
        pSM = p00;
END CASE;

%===================================================================
CBR DRAM Refresh State Machine
===================================================================%

cbrSM.clk           = clk36_r;
cbrSM.reset         = !clr_l;

CASE cbrSM IS
```

```
    WHEN cIdle =>             -- <nothing> active
        IF docbr_h THEN cbrSM = cCBR1;
        END IF;
    WHEN cCBR1 =>             -- wrallcas_h active
        cbrSM = cCBR2;
    WHEN cCBR2 =>             -- wrallcas_h, wrallras_h active
        cbrSM = cCBR3;
    WHEN cCBR3 =>             -- wrallcas_h, wrallras_h active
        cbrSM = cCBR4;
    WHEN cCBR4 =>             -- wrallras_h, cbrdone_h active
        cbrSM = cIdle;        -- rfshreq_h inactive on next cycle
END CASE;

%===========================================================================
DRAM Write State Machine
===========================================================================%
dramSM.clk       = clk36_r;
dramSM.reset     = !clr_l;

-- 'canras_h' is active when the DRAM SM CAN transition into state dRas on next cycle.
-- Whether or not is does so is dependent on whether there is data to write (ie.
-- 'startcycle_h' is active. LCELL needed to eliminate sharable expanders in
-- equation for 'enrfren_h'.
canras_h         = ((dramSM == dIdle1) # (dramSM == dIdle2)) & !rfshreq_h;    -- LCELL

-- 'willcas_h' is active when the DRAM SM WILL transition into state dCas on next cycle.
-- LCELL needed to eliminate sharable expanders in equation for 'enrfren_h'.
willcas_h        = (dramSM == dColSel);                                       -- LCELL

-- 'stop_h' is active when the DRAM SM is in state dCas and it is determined that
-- the machine will NOT go back into dColSel to continue a fast page mode write.
-- LCELL needed to eliminate sharable expanders in equation for 'enrfren_h'.
stop_h           = (dramSM == dCas) & (endline_h # endrow_h # rfshreq_h);

-- Precursors (signals that go active the clock BEFORE their namesake signals go active)
-- "!wrmuxsel_h[2]" is equivalent to "msred # msgreen # msblue"
pc_count_h       = willcas_h & (!wrmuxsel_h[2] # (wrmuxsel_h[] == msmsblue));

-- 'docount_h' goes active every time we enter state dCas
docount_h.clk    = clk36_r;
!docount_h.clrn  = !clr_l;
docount_h.d      = willcas_h;

-- 'count_h' goes active when we want to increment 'wradrcnt_h' and decrement 'wrpixcnt_h'
count_h.clk      = clk36_r;
!count_h.clrn    = !clr_l;
count_h.d        = pc_count_h;

-- 'load_h' goes active when we want to load 'wrpixcnt_h[]'
load_h.clk       = clk36_r;
!load_h.clrn     = !clr_l;

CASE dramSM IS
    WHEN dIdle1 =>            -- <nothing> active
        IF rfshreq_h THEN dramSM = dRfsh1;
        ELSIF startcycle_h THEN dramSM = dRas; load_h.d = VCC;
        END IF;
    WHEN dRfsh1 =>            -- docbr_h active
        IF cbrdone_h THEN dramSM = dIdle1;
        END IF;

    WHEN dRas =>              -- wrras_h active, (load_h.q) may be active
        dramSM = dColSel;
    WHEN dColSel =>           -- wrras_h, we_l, colsel_h active
```

```
        dramSM = dCas;
    WHEN dCas =>              -- wrras_h, wrcas_h, we_l, colsel_h (docount_h) active
        IF endline_h THEN dramSM = dNewLn;
        ELSIF rfshreq_h THEN dramSM = dRfsh2;
        ELSIF !endrow_h & willgo_h THEN dramSM = dColSel;
        ELSE dramSM = dRasPre;
        END IF;

    WHEN dNewLn =>            -- <nothing> active
        dramSM = dWait;
    WHEN dWait =>             -- <nothing> active
        dramSM = dAdd;
    WHEN dAdd =>              -- add_h active
        dramSM = dIdle1;

    WHEN dRasPre =>           -- <nothing> active
        dramSM = dIdle2;
    WHEN dIdle2 =>            -- <nothing> active
        IF !xfren_h THEN dramSM = dIdle1;
        ELSIF rfshreq_h THEN dramSM = dRfsh2;
        ELSIF startcycle_h THEN dramSM = dRas;
        END IF;
    WHEN dRfsh2 =>            -- docbr_h active
        IF cbrdone_h THEN dramSM = dIdle2;
        END IF;
END CASE;

-- RGB Selector
rgbsel_h[].clk       = clk36_r;
!rgbsel_h[].clrn     = !clr_l;
IF load_h # add_h THEN
    rgbsel_h[].d     = 0;
ELSIF docount_h THEN
    rgbsel_h[].d     = rgbsel_h[].q + 1;
ELSE
    rgbsel_h[].d     = rgbsel_h[].q;
END IF;

-- Super Mux Control Signals
wrmuxsel_h[2]        = (mode_h[] == rgbmode);
IF mode_h[] == rgbmode THEN
    wrmuxsel_h[1..0] = rgbsel_h[];
ELSE
    wrmuxsel_h[1..0] = mode_h[1..0];
END IF;

-- Video DRAM Multiplexor
CASE colsel_h IS
    WHEN B"0" =>
        wra_h[] = wradrcnt_h[19..10].q;
    WHEN B"1" =>
        wra_h[] = wradrcnt_h[09..00].q;
END CASE;

%=====================================================================
Read FIFO Data Shift Registers
=====================================================================%

-- First Stage
rfsca_h.clk          = clk36_r;       -- cmd / ~data bit
!rfsca_h.clrn        = !clr_l;
rfsca_h.ena          = cdavail_h;
rfsca_h.d            = rfsc_h;

rfda_h[].clk         = clk36_r;
```

```
    !rfda_h[].clrn      = !clr_l;
    rfda_h[].ena        = cdavail_h;
    rfda_h[].d          = rfd_h[];

    -- Second Stage
    rfdb_h[].clk        = clk36_r;
    !rfdb_h[].clrn      = !clr_l;
    rfdb_h[].ena        = cdavail_h;
    rfdb_h[].d          = rfda_h[].q;

    -- Third Stage
    rfdc_h[].clk        = clk36_r;
    !rfdc_h[].clrn      = !clr_l;
    rfdc_h[].ena        = cclken_h;
    rfdc_h[15..08].d    = rfda_h[].q;
    rfdc_h[07..00].d    = rfdb_h[].q;

    wrd_h[]             = rfdc_h[].q;

    %=====================================================================
    LED Status Outputs

    We employ pulse stretching counters to be able to see the LED.
    =====================================================================%
    rbusycnt_h[].clk    = ckr_r;
    !rbusycnt_h[].clrn  = !clr_l;
    IF !rfwen_l THEN
        rbusycnt_h[].d = 15;
    ELSIF rbusycnt_h[] != 0 THEN
        rbusycnt_h[].d = rbusycnt_h[].q - 1;
    ELSE
        rbusycnt_h[].d = rbusycnt_h[].q;
    END IF;
    !rbusy_l     = (rbusycnt_h[] != 0) & rcd_h;

    tbusycnt_h[].clk    = 33mhz_r;
    !tbusycnt_h[].clrn  = !clr_l;
    IF twr_h THEN
        tbusycnt_h[].d = 15;
    ELSIF tbusycnt_h[] != 0 THEN
        tbusycnt_h[].d = tbusycnt_h[].q - 1;
    ELSE
        tbusycnt_h[].d = tbusycnt_h[].q;
    END IF;
    !tbusy_l     = (tbusycnt_h[] != 0);

END; % WRITE.TDF %
```

KENSAL CORPORATION TRADE SECRETS

Motor Control Board Register Definitions

Revision A: November 26, 1997

Motor Control Board - Base Address Register Assignments

BAR	Description
0	AMCC 5933Q Matchmaker PCI Operation Registers (64 bytes)
1	Altera Controller (16 bytes)

Altera Controller Assignments

Addr	Name	Special Conditions
0x0C	Full/Micro Step Limit Register	Write/Readback
0x08	Parallel Output Register	Write/Readback
0x04	Limit Register	Read only
0x00	Control/Status Register	Some bits read only

Control/Status Register

This read/write register enables interaction with the PMD motion control chipsets. It also allows control over interrupt propagation.

Limit Register

This read-only register contains raw limit switch inputs as well as combinatorially processed limit outputs that go the MC1241A chipsets.

Parallel Output Register

This read/write register contains PC microscope specific as well as undedicated output bits.

Full/Micro Step Limit Register

This read/write register contains two counter limit values, fulllimit_h[13..00] and microlimit_h[13..00] used to switch the mode of each axis on-the-fly between microstepping and full step.

Control/Status Register Bit Definitions

Bit	Name	Write	Read
31			
30			
29			
28			
27	fulltravel_h	Allow Full Travel	Allow Full Travel
26	photointena_h	Photo Sensor Interrupt Enable	Photo Sensor Interrupt Enable
25	photoint_h	<0 -> clears bit, 1 -> NOP>	Photo Sensor Interrupt
24	hostintena_h[1]	Host Interrupt Enable 1	Host Interrupt Enable 1
23	hostintena_h[0]	Host Interrupt Enable 0	Host Interrupt Enable 0
22	hostint_h[1]	Host Interrupt 1	Host Interrupt 1
21	hostint_h[0]	Host Interrupt 0	Host Interrupt 0
20	hostrdy_h[1]	<read only)	Host Ready 1
19	hostrdy_h[0]	<read only)	Host Ready 0
18	mio_h[2]	Motor I/O Operation Bit 2	<cleared after the operation>
17	mio_h[1]	Motor I/O Operation Bit 1	<cleared after the operation>
16	mio_h[0]	Motor I/O Operation Bit 0	<cleared after the operation>
15	mdat_h[15]	Motor Data Bit 15	Motor Data Bit 15
14	mdat_h[14]	Motor Data Bit 14	Motor Data Bit 14
13	mdat_h[13]	Motor Data Bit 13	Motor Data Bit 13
12	mdat_h[12]	Motor Data Bit 12	Motor Data Bit 12
11	mdat_h[11]	Motor Data Bit 11	Motor Data Bit 11
10	mdat_h[10]	Motor Data Bit 10	Motor Data Bit 10
09	mdat_h[09]	Motor Data Bit 9	Motor Data Bit 9
08	mdat_h[08]	Motor Data Bit 8	Motor Data Bit 8
07	mdat_h[07]	Motor Data Bit 7	Motor Data Bit 7
06	mdat_h[06]	Motor Data Bit 6	Motor Data Bit 6
05	mdat_h[05]	Motor Data Bit 5	Motor Data Bit 5
04	mdat_h[04]	Motor Data Bit 4	Motor Data Bit 4
03	mdat_h[03]	Motor Data Bit 3	Motor Data Bit 3
02	mdat_h[02]	Motor Data Bit 2	Motor Data Bit 2
01	mdat_h[01]	Motor Data Bit 1	Motor Data Bit 1
00	mdat_h[00]	Motor Data Bit 0	Motor Data Bit 0

mdat_h[15..00] Motor Data

These bits are used to transfer data to/from the selected PMD motor controller. After a data read operation is processed (see mio_h[] below), the read data will be available by reading these bits. Data should be present in these bits when the data write operation is performed. When a command write operation is issued only bits [07..00] are used. On poweron or global reset, these bits are set to 0.

mio_h[2..0] Motor I/O Operation

These bits control the interface with the PMD motor controller IC's. There are two independent MC1241A chipsets on the MCB. Each chipset controls 2 motors. Thus, the board supports up to 4 axis of motion control. Motors #0 and #1 are controlled by chipset #0. Motors #2 and #3 are controlled by chipset #1. Three operations are available for each chip set: 1) Command Write, 2) Data Write, and 3) Data Read. 8-bits are sent with a Command Write and 16-bits are transferred for a Data Read/Write. The bit descriptions are:

mio_h[2..0]	Description
111	Data Write to chip set #1
110	Data Read from chip set #1
101	Command Write to chip set #1
100	No operation
011	Data Write to chip set #0
010	Data Read from chip set #0
001	Command Write to chip set #0
000	No operation

These bits are reset after the requested operation with the selected chipset is performed. This happens so fast that a non-zero write followed immediately by a read would probably yield a value of 0. On poweron or global reset, these bits are set to 0.

hostrdy_h[1..0] Host Ready

Before any I/O operation is performed to a particular motor controller chipset, the driver must insure that the chipset to be addressed is ready to receive the data. These bits are connected directly to the HostRdy bits of the respective controllers. Immediately after performing I/O to a chipset, the hostrdy_h[] bit of the respective controller will go inactive for about 12 microseconds. During this time, the controller is unable to receive or transmit commands/data.

hostint_h[1..0] Host Interrupt

Each chipset has the capability generating an interrupt (see the MC1241A data sheet for details). If the particular chipset is generating an interrupt, the corresponding bit will be active. In order to clear the interrupt, the interrupting MC1241A chipset must be interrogated to determine the cause of the interrupt(s). A “RST_INTRPT” command is then sent to the chipset to clear the interrupt(s). In order for the interrupt to be propagated to the PCI bus, the corresponding interrupt enable bit (see below) must be set. On poweron or global reset, these bits are set to 0.

hostintena_h[1..0] Host Interrupt Enable

These bits enable the interrupt generated by a particular chipset to propagate onto the PCI bus. We will use the mailbox technique to get the interrupt onto the bus. Some details may be found in the Spring 1996 “S5933 PCI Controller Data Book”, section 9.1.2, beginning on page 9-4. The S5933 may be programmed to trigger an interrupt when a particular incoming mailbox is written to from the add-on side. In our case, we will use mailbox #4, byte #0. Bit 0 (ie. 0x00000001) will be set if chipset #0 is interrupting - bit 1 (ie. 0x00000002) for chipset #1. To prevent race conditions, the Motor Control board will continually write to mailbox #4 whenever an interrupt is active. Because of this, to clear the interrupt(s) you must first clear the appropriate interrupt bits (‘hostint_h[1..0]’) by interacting with the motor control chipsets, then read mailbox #4 to clear the PCI interrupt. Ordering is important here - if you don’t clear the source of the interrupt, the mailbox will generate interrupts continuously. The last values written may be read back. On poweron or global reset, these bits are set to 0.

photoint_h Photo Sensor Interrupt

This bit becomes set when the rising edge of photosensor_h is detected. It is cleared by writing a ‘0’ to it. Writing a ‘1’ has no effect. In order for the interrupt to be propagated to the PCI bus, bit photointena_h (see below) must be set. On poweron or global reset, this bit is set to 0.

photointena_h Photo Sensor Interrupt Enable

This bit enables the photoint_h interrupt to propagate onto the PCI bus. More precisely, it results in a value of 0x00000004 to be written to mailbox #4, byte #0 (see the discussion for the hostinten_h[] bits above). The last value written may be read back. On poweron or global reset, this bit is set to 0.

fulltravel_h Allow Full Travel

When reset, this bit will curtail travel in the positive 'Y' direction. It does this by imposing a positive 'Y' limit when out of the negative 'Y' limit longer than 2 milliseconds. It is provided as an additional safeguard to prevent unintentional stage travel from damaging the optics within the enclosure. After the software has successfully found the home position and backed out of the load channel, it should activate this bit to allow for unrestricted travel. On poweron or global reset, this bit is set to 0.

Limit Register Bit Definitions

Bit	Name	Write	Read
31			
30			
29			
28			
27			
26			
25			
24			
23	negwlimit_h	<read only>	Motor #3 Negative Limit
22	poswlimit_h	<read only>	Motor #3 Positive Limit
21	negzlimit_h	<read only>	Motor #2 Negative Limit
20	poszlimit_h	<read only>	Motor #2 Positive Limit
19	negylimit_h	<read only>	Motor #1 Negative Limit
18	posylimit_h	<read only>	Motor #1 Positive Limit
17	negxlimit_h	<read only>	Motor #0 Negative Limit
16	posxlimit_h	<read only>	Motor #0 Positive Limit
15	scandislimit_h	<read only>	Scan Disable Limit
14	scanenalimit_h	<read only>	Scan Enable Limit
13	fortyxlimit_h	<read only>	40x Lens Limit
12	tenxlimit_h	<read only>	10x Lens Limit
11	sparelimit_h3	<read only>	Spare Limit Bit 3
10	photosensor_h	<read only>	Photo Sensor
09	nzlimit_h	<read only>	-Z Limit
08	pzlimit_h	<read only>	+Z Limit
07	sparelimit_h2	<read only>	Spare Limit Bit 2
06	sparelimit_h1	<read only>	Spare Limit Bit 1
05	nylimit_h	<read only>	-Y Limit
04	pylimit_h	<read only>	+Y Limit
03	sparelimit_h0	<read only>	Spare Limit Bit 0
02	ldlimit_h	<read only>	Load Limit
01	nxlimit_h	<read only>	-X Limit
00	pxlimit_h	<read only>	+X Limit

pxlimit_h, nxlimit_h ±X Limits
pylimit_h, nylimit_h ±Y Limits
pzlimit_h, nzlimit_h ±Z Limits
ldlimit_h Load Limit

These bits are raw inputs from the OSC PC microscope for the various travel limit sensors. They are processed by the Altera CPLD to form motor limits for the X, Y, and Z motor axis (explained below). Active bits signify exceeded limits.

tenxlimit_h, fortyxlimit_h 10x and 40x Limits

These bits are from sensors on the lens selector. When 'tenxlimit_h' is active, the lens is in the 10x position. When 'fortyxlimit_h' is active, the lens is in the 40x position. When neither is active, the lens is somewhere in between.

scanenalimit_h, scandislimit_h Scan Enable and Disable Limits

Similar to the 10x and 40x limits, these bits signal whether or not the RL4000P is pressed against the slide ('scanenalimit_h' active). Signal 'scandislimit_h' will be active when the RL4000P is in the retracted position. If neither signal is active, the sensor is in an unknown position.

sparelimit_h[3..0] Spare Limits

These bits are undedicated inputs from test points on the Motor Control PCB. In addition to being readable by this register, they are fed into the Altera CPLD, which combinatorially creates the actual limits used for the four motors controlled by this board.

posxlimit_h, negxlimit_h Motor #0 Positive and Negative Limits
posylimit_h, negylimit_h Motor #1 Positive and Negative Limits
poszlimit_h, negzlimit_h Motor #2 Positive and Negative Limits
poswlimit_h, negwlimit_h Motor #3 Positive and Negative Limits

Information from various sensor inputs (bits [15..00] of this register) are fed into the Altera CPLD. Combinatorial circuitry within the chip is used to "process" this data to prepare the actual motor limits used by the motor controllers. In this way, non-rectangular limit courtyards can be defined. In addition to feeding the motor controllers, the states of these bits can be read by this register.

Parallel Output Register Bit Definitions

Bit	Name	Write	Read
31			
30			
29			
28			
27	overridew_h	Override W-Axis Driver	Override W_Axis Driver
26	overridez_h	Override Z-Axis Driver	Override Z_Axis Driver
25	overridey_h	Override Y-Axis Driver	Override Y_Axis Driver
24	overridex_h	Override X-Axis Driver	Override X_Axis Driver
23	brakew_h	Brake W-Axis Driver	Brake W-Axis Driver
22	brakez_h	Brake Z-Axis Driver	Brake Z-Axis Driver
21	brakey_h	Brake Y-Axis Driver	Brake Y-Axis Driver
20	brakex_h	Brake X-Axis Driver	Brake X-Axis Driver
19	enablew_h	Enable W-Axis Driver	Enable W-Axis Driver
18	enablez_h	Enable Z-Axis Driver	Enable Z-Axis Driver
17	enabley_h	Enable Y-Axis Driver	Enable Y-Axis Driver
16	enablex_h	Enable X-Axis Driver	Enable X-Axis Driver
15	bufout_h7	Buffered Output Bit 7	Buffered Output Bit 7
14	bufout_h6	Buffered Output Bit 6	Buffered Output Bit 6
13	bufout_h5	Buffered Output Bit 5	Buffered Output Bit 5
12	bufout_h4	Buffered Output Bit 4	Buffered Output Bit 4
11	bufout_h3	Buffered Output Bit 3	Buffered Output Bit 3
10	bufout_h2	Buffered Output Bit 2	Buffered Output Bit 2
09	bufout_h1	Buffered Output Bit 1	Buffered Output Bit 1
08	bufout_h0	Buffered Output Bit 0	Buffered Output Bit 0
07	ocout_h1	Open Collector Output #1	Open Collector Output #1
06	ocout_h0	Open Collector Output #0	Open Collector Output #0
05	spare_h1	Spare 1 (test point and grn LED)	Spare 1 (test point and grn LED)
04	spare_h0	Spare 0 (test point and red LED)	Spare 0 (test point and red LED)
03	scanena_h	Scanning Select Enable	Scanning Select Enable
02	scansel_h	Scanning Select	Scanning Select
01	tenxena_h	10x Lens Select Enable	10x Lens Select Enable
00	tenxsel_h	10x Lens Select	10x Lens Select

tenxsel_h, tenxena_h 10x Select and Enable

A bi-directional solenoid is used to select between 10x and 40x lenses. The solenoid is designed for momentary actuation, holding the previously selected state until a new state is desired. The 'tenxena_h' bit turns the solenoid on. If bit 'tenxsel_h' is active, the activated solenoid will move toward the 10x lens position. If 'tenxsel_h' is inactive, the solenoid will move toward the 40x position. The transition should take approximately 50ms, at which time bit 'tenxena_h' should be made inactive, which turns the solenoid off. Limit sensors 'fortyxlimit_h' and 'tenxlimit_h' from the Limit Register may be used to monitor the progress of the lens change. If the solenoid is left 'on' damage may occur, so steps should be made to insure that it never stays on longer than, say, 500ms. The last value written may be read back. On poweron or global reset, these bits are set to 0b00.

scansel_h, scanena_h Scan Select and Enable

These two bits function identically to bits 'tenxsel_h' and 'tenxena_h' except these bits affect the loading and unloading of the RL4000P lensless scanner. To recap, if 'scansel_h' and 'scanena_h' are active, the solenoid will move the RL4000P sensor to be in contact with the slide. An inactive 'scansel_h' will retract the sensor. When the desired movement is complete, the solenoid is shut off by deactivating 'scanena_h'. Limit sensors 'scanenalimit_h' and 'scandislimit_h' may be used to monitor the application/retraction of the scanner. Like 10x solenoid above, precautions should be taken to insure that the solenoid is not left on. The last value written may be read back. On poweron or global reset, these bits are set to 0b00.

spare_h0, spare_h1 Spare Bits

These bits are undedicated outputs that go to test points and LEDs. When 'spare_h0' is active a red LED will activate and the test point will go to a low logic level. 'spare_h1' activates a green LED. The last value written may be read back. On poweron or global reset, these bits are set to 0b00.

ocout_h0, ocout_h1 Open Collector Outputs

These bits are undedicated high current (500 mA) open collector (actually, open drain) outputs. They are pulled up to +5V with 10K resistors and go to test points as well as a connector. The last value written may be read back. On poweron or global reset, these bits are set to 0b00.

bufout_h[7..0] Buffered Outputs

These bits are undedicated buffered (via 74ABT244) outputs. They terminate at a connector leaving the board. The last value written may be read back. On poweron or global reset, these bits are set to 0b00000000.

enablex_h, enabley_h, enablez_h, enablew_h Enable Driver

These bits connect directly to the enable_l pins of the PWM motor drivers. If these bits are zero, the all motor driver output transistors shut off, allowing the motors to spin as if unconnected. On poweron or global reset, these bits are set to 0b0000, so it will be necessary to write a '1' to an appropriate bit to use a particular axis.

brakex_h, brakey_h, brakez_h, brakew_h Brake Driver

These bits connect directly to the brake_l pins of the PWM motor drivers. They are not normally used and are included mainly for completeness. On poweron or global reset, these bits are set to 0b0000.

overridex_h, overridey_h, overridez_h, overridew_h Override Driver

These bits control logic within the Altera chip which then results in the output of the mode_h[] signals to the PWM motor drivers. When set, an override forces fast decay microstep mode. These bits are not normally used but are included for completeness. On poweron or global reset, these bits are set to 0b0000.

Full/Micro Step Limit Register Bit Definitions

Bit	Name	Write	Read
31			
30			
29	fulllimit_h13	Full Limit Bit 13	Full Limit Bit 13
28	fulllimit_h12	Full Limit Bit 12	Full Limit Bit 12
27	fulllimit_h11	Full Limit Bit 11	Full Limit Bit 11
26	fulllimit_h10	Full Limit Bit 10	Full Limit Bit 10
25	fulllimit_h09	Full Limit Bit 09	Full Limit Bit 09
24	fulllimit_h08	Full Limit Bit 08	Full Limit Bit 08
23	fulllimit_h07	Full Limit Bit 07	Full Limit Bit 07
22	fulllimit_h06	Full Limit Bit 06	Full Limit Bit 06
21	fulllimit_h05	<don't care>	0
20	fulllimit_h04	<don't care>	0
19	fulllimit_h03	<don't care>	0
18	fulllimit_h02	<don't care>	0
17	fulllimit_h01	<don't care>	0
16	fulllimit_h00	<don't care>	0
15			
14			
13	microlimit_h13	Micro Limit Bit 13	Micro Limit Bit 13
12	microlimit_h12	Micro Limit Bit 12	Micro Limit Bit 12
11	microlimit_h11	Micro Limit Bit 11	Micro Limit Bit 11
10	microlimit_h10	Micro Limit Bit 10	Micro Limit Bit 10
09	microlimit_h09	Micro Limit Bit 09	Micro Limit Bit 09
08	microlimit_h08	Micro Limit Bit 08	Micro Limit Bit 08
07	microlimit_h07	Micro Limit Bit 07	Micro Limit Bit 07
06	microlimit_h06	Micro Limit Bit 06	Micro Limit Bit 06
05	microlimit_h05	<don't care>	0
04	microlimit_h04	<don't care>	0
03	microlimit_h03	<don't care>	0
02	microlimit_h02	<don't care>	0
01	microlimit_h01	<don't care>	0
00	microlimit_h00	<don't care>	0

fulllimit_h[13..00], microlimit_h[13..00]   Full/Micro Limit Registers

The Motor Control Board has the unique capability to switch, on-the-fly, between microstep and full step operations. During microstep operation, each physical motor step is divided into 64 smaller micro steps. Thus, with the 400 steps/rev. motors we are using, our resolution is increased to 400 x 64 or 25,600 steps per revolution. Another added benefit is smoother low speed operation while in microstep mode. As motor speed increases however, the microstepping waveform supplied to the motor produces insufficient torque to drive the motor faster. Full step driving increases this torque, thus, we'd like to switch to full step mode as the motor speeds up and switch back to microstepping for higher resolution and smoother operation when the motor slows down.

The fulllimit_h[] value is defined by the following equation:

<step freq. when fullstep mode is entered> = 33.33 x 10^6 / fulllimit_h[]

The microlimit_h[] value is defined by the following equation:

<step freq. when microstep mode is entered> = 33.33 x 10^6 / microlimit_h[]

For proper operation, fulllimit_h[] should be smaller than microlimit_h[]. Typical default values for these two registers are 0x3580 for fulllimit_h[] and 0x3F00 for microlimit_h[]. These values will cause transition to fullstep mode at ~6 rps (revolutions per second), and a transition back to microstep mode at ~5.1 rps.

If you'd like the controller never to enter full step mode, set fulllimit_h[] to 0x0040. Since you'll never get to full step mode, the value stored in microlimit_h[] is ignored.

```
%==========================================================================
PCI Motor Controller

Kensal Corporation
OSC Project

Author: Ken Crocker
Date:   19 Sep 97
File:   EDGEDET.TDF
Rev:    1.0

Revision History:
    1.1      1/2/98       Removed 'clr_l' from F/F async clear.
==========================================================================%
TITLE "PCI Motor Controller - Edge Detector";

SUBDESIGN edgedet
(
    clk_r               : INPUT;        -- 10 kHz system clock
    clr_l               : INPUT;        -- active low asynchronous system reset

    in_h                : INPUT;
    out_h               : OUTPUT;
)

VARIABLE
    in_h1               : DFF;
    in_h2               : DFF;

--  out_h               : DFF;

BEGIN
    in_h1.clk       = clk_r;
--  !in_h1.clrn     = !clr_l;
    in_h1.d         = in_h;

    in_h2.clk       = clk_r;
--  !in_h2.clrn     = !clr_l;
    in_h2.d         = in_h1;

--  out_h.clk       = clk_r;
--  !out_h.clrn     = !clr_l;
--  out_h.d         = in_h1 $ in_h2;
    out_h           = in_h1 $ in_h2;
END;
```

```
%==========================================================================
PCI Motor Controller
Full/Micro Step Selector

Kensal Corporation
OSC Project

Author: Ken Crocker
Date:   16 Oct 97
File:   FULLSTEP.TDF
Rev:    1.0
==========================================================================%
TITLE "PCI Full/Micro Step Selector";

INCLUDE "edgedet.inc";

SUBDESIGN fullstep
(
    clk_r                   : INPUT;
    clr_l                   : INPUT;

    -- Full/Micro Limit Registers
    fulllimit_h[7..0]       : INPUT;
    microlimit_h[7..0]      : INPUT;
    override_h              : INPUT;    -- forces microstep, fast decay mode
    brake_l                 : INPUT;    -- forces braking, fast decay mode

    -- Inputs from Motor Driver
    phase_h[1..0]           : INPUT;

    -- Output to Full/Micro Control Ckt
    mode_h[1..0]            : OUTPUT;
    selfull_h[1..0]         : OUTPUT;
)

VARIABLE
    -- Full/Micro Step Operations
    edgedet1                : edgedet;
    edgedet0                : edgedet;
    mode_h[1..0]            : SRFF;
    selfull_h[1..0]         : SRFF;
    edgedet_h               : DFF;
    cnt_h[13..00]           : DFF;
    cnttcls_h               : DFF;
    cnttc_h                 : LCELL;

    moSM                    : MACHINE OF BITS (
                                fullstep_h
                            ) WITH STATES (
                                mo00 = B"0",
                                mo01 = B"0",
                                mo02 = B"1"
                            );

BEGIN
    edgedet1.(clk_r, clr_l, in_h) = (clk_r, clr_l, phase_h[1]);
    edgedet0.(clk_r, clr_l, in_h) = (clk_r, clr_l, phase_h[0]);

    mode_h[].clk        = clk_r;    -- high -> fast decay mode
    !mode_h[].clrn      = !clr_l;

    -- 'override_h' forces slow decay mode
--  mode_h[1].s         = edgedet0.out_h & !override_h  # !brake_l;
--  mode_h[1].r         = (edgedet1.out_h #  override_h) &  brake_l;
```

```
--  mode_h[0].s         =  edgedet1.out_h & !override_h  # !brake_l;
--  mode_h[0].r         = (edgedet0.out_h #  override_h) &  brake_l;

    -- mode_h[] high -> fast decay mode
    -- Input priorities:
    --      1) '!brake_l' forces fast decay mode
    --      2) 'selfull_h[]' forces slow decay mode
    --      3) 'override_h' forces fast decay mode
    --
    -- mode_h[1].s  edgedet0.out    override_h  selfull_h[1]    brake_l
    --      1           X               X           X              0
    --      0           X               X           1              1
    --      1           X               1           0              1
    --      0           0               0           0              1
    --      1           1               0           0              1
    --
    -- mode_h[1].r  edgedet1.out    override_h  selfull_h[1]    brake_l
    --      x           0               0           0              0
    --      x           0               0           0              1
    --      x           0               0           1              0
    --      x           0               0           1              1
    --      x           0               1           0              0
    --      x           0               1           0              1
    --      x           0               1           1              0
    --      x           0               1           1              1
    --      x           1               0           0              0
    --      x           1               0           0              1
    --      x           1               0           1              0
    --      x           1               0           1              1
    --      x           1               1           0              0
    --      x           1               1           0              1
    --      x           1               1           1              0
    --      x           1               1           1              1
    mode_h[1].s         = ((edgedet0.out_h #  override_h) & !selfull_h[1]) # !brake_l;
    mode_h[1].r         =   edgedet1.out_h & !override_h  &  brake_l # selfull_h[1];
    mode_h[0].s         = ((edgedet1.out_h #  override_h) & !selfull_h[0]) # !brake_l;
    mode_h[0].r         =   edgedet0.out_h & !override_h  &  brake_l # selfull_h[0];

    -- 'override_h' forces microstep mode
    selfull_h[].clk     = clk_r;
    !selfull_h[].clrn   = !clr_l;
    selfull_h[1].s      =  fullstep_h & edgedet0.out_h & !override_h;
    selfull_h[1].r      = !fullstep_h & edgedet0.out_h # override_h;
    selfull_h[0].s      =  fullstep_h & edgedet1.out_h & !override_h;
    selfull_h[0].r      = !fullstep_h & edgedet1.out_h # override_h;

    edgedet_h.clk       = clk_r;
    !edgedet_h.clrn     = !clr_l;
    edgedet_h.d         = edgedet1.out_h # edgedet0.out_h;

    cnt_h[].clk = clk_r;
    !cnt_h[].clrn    = !clr_l;

    -- cnt_h[05..00] (break into 2 parts to reduce fan-in)
    IF edgedet_h THEN
        cnt_h[05..00].d = B"000000";
    ELSE
        cnt_h[05..00].d = cnt_h[05..00].q - 1;
    END IF;
    cnttcls_h.clk    = clk_r;
    !cnttcls_h.clrn = !clr_l;
    cnttcls_h.d      = !edgedet_h & (cnt_h[05..00].q == 1);

    -- cnt_h[13..06]
```

```
    IF edgedet_h THEN
        IF LCELL (moSM == mo00) THEN
--          cnt_h[07..00].d = fulllimit_h[];     -- simulation
            cnt_h[13..06].d = fulllimit_h[];     -- smaller number (higher rpm limit)
        ELSE
--          cnt_h[07..00].d = microlimit_h[];    -- simulation
            cnt_h[13..06].d = microlimit_h[];    -- big number (lower rpm limit)
        END IF;
    ELSIF cnttcls_h THEN
        cnt_h[13..06].d = cnt_h[13..06].q - 1;
    ELSE
        cnt_h[13..06].d = cnt_h[13..06].q;
    END IF;
    cnttc_h      = (cnt_h[].q == 0);             -- LCELL

    moSM.clk     = clk_r;
    moSM.reset   = !clr_l;
    CASE moSM IS
        -- Motor is in microstep mode. Waiting for edge in order to take a measurement
        WHEN mo00 =>        -- <nothing> active
            IF edgedet_h THEN moSM = mo01;
            END IF;
        -- Waiting for another phase edge to complete measurement
        WHEN mo01 =>        -- <nothing> active
            IF cnttc_h THEN moSM = mo00;          -- counter expired first. Motor running slower than
fulllimit
            ELSIF edgedet_h THEN moSM = mo02;     -- edge detected first. Motor running faster than
fulllimit
            END IF;

        -- Motor is in fullstep mode. Taking phase frequency measurment
        WHEN mo02 =>        -- fullstep_h active
            IF cnttc_h THEN moSM = mo00;          -- counter expired first. Motor running slower than
microlimit
            END IF;                               -- edge detected first. Motor running faster than
microlimit

    END CASE;

END;
```

```
%===========================================================================
PCI Motor Controller (Top Level)

Kensal Corporation
OSC Project

Author: Ken Crocker
Date:   8 Nov 97
File:   MOTOR.TDF
Rev:    1.0
===========================================================================%
TITLE "PCI Motor Controller - Top Level";

INCLUDE "fullstep.inc";

-- Values for "ptnum_h[1..0]"
CONSTANT motorCtrl       = B"00";

-- Longword offset within passthru region
CONSTANT motCtrlStat     = B"0000";  -- 0x00 >> 2 = 0x0
CONSTANT motLimit        = B"0001";  -- 0x04 >> 2 = 0x1
CONSTANT motOutput       = B"0010";  -- 0x08 >> 2 = 0x2
CONSTANT motFullMicro    = B"0011";  -- 0x0C >> 2 = 0x3

-- adr_h[6..2] Constants
CONSTANT AOMB4           = B"00111"; -- 0x1C >> 2 = 0x07
CONSTANT APTD            = B"01011"; -- 0x2C >> 2 = 0x0B

SUBDESIGN motor
(
    bpclk_r                : INPUT;        -- buffered PCI clock
    gclk2_r                : INPUT;        -- reserved
    sysrst_l               : INPUT;        -- system reset
    goe_l                  : INPUT;        -- reserved

    -- AMCC S5933 Passthru Interface Signals
    ptatn_l                : INPUT;     -- pass thru cycle input
    ptburst_l              : INPUT;     -- burst access input
    ptnum_h[1..0]          : INPUT;     -- base address register number
    ptbe_l[3..0]           : INPUT;     -- requested byte enables
    ptwr_h                 : INPUT;     -- write (ptrd_l) input
    ptadr_l                : OUTPUT;    -- address reg select (slow slew rate)
    ptrdy_l                : OUTPUT;    -- ready output (slow slew rate)

    -- AMCC S5933 Add-On Bus Interface Signals
    dq_h[31..00]           : BIDIR;     -- data bus (slow slew rate)
    adr_h[6..2]            : OUTPUT;    -- register address (slow slew rate)
    select_l               : OUTPUT;    -- cycle start (slow slew rate)
    rd_l                   : OUTPUT;    -- read strobe (slow slew rate)
    wr_l                   : OUTPUT;    -- write strobe (slow slew rate)

    -- Interface to Stepper Motor Controllers
    hostrdy_h[1..0]        : INPUT;
    hostint_l[1..0]        : INPUT;
    hostslct_l[1..0]       : OUTPUT;
    hostcmd_h[1..0]        : OUTPUT;
    hostwrite_l[1..0]      : OUTPUT;
    hostread_l[1..0]       : OUTPUT;

    -- Interface to stepper motor drivers
    phasex_h[1..0]         : INPUT;
    phasey_h[1..0]         : INPUT;
    phasez_h[1..0]         : INPUT;
    phasew_h[1..0]         : INPUT;
```

```
    selfullx_h[1..0]        : OUTPUT;
    selfully_h[1..0]        : OUTPUT;
    selfullz_h[1..0]        : OUTPUT;
    selfullw_h[1..0]        : OUTPUT;
    modex_h[1..0]           : OUTPUT;
    modey_h[1..0]           : OUTPUT;
    modez_h[1..0]           : OUTPUT;
    modew_h[1..0]           : OUTPUT;
    enablex_l               : OUTPUT;
    enabley_l               : OUTPUT;
    enablez_l               : OUTPUT;
    enablew_l               : OUTPUT;
    brakex_l                : OUTPUT;
    brakey_l                : OUTPUT;
    brakez_l                : OUTPUT;
    brakew_l                : OUTPUT;

    -- Interface to OSC limit switches
    pxlimit_h               : INPUT;
    nxlimit_h               : INPUT;
    pylimit_h               : INPUT;
    nylimit_h               : INPUT;
    pzlimit_h               : INPUT;
    nzlimit_h               : INPUT;
    ldlimit_h               : INPUT;
    tenxlimit_h             : INPUT;
    fortyxlimit_h           : INPUT;
    scanenalimit_h          : INPUT;
    scandislimit_h          : INPUT;
    photosensor_h           : INPUT;
    sparelimit_h[3..0]      : INPUT;

    -- Interface to OSC Actuators / Parallel Port Outputs
    tenxsel_h               : OUTPUT;
    tenxena_l               : OUTPUT;
    scansel_h               : OUTPUT;
    scanena_l               : OUTPUT;
    spare_l[1..0]           : OUTPUT;
    oc_h[1..0]              : OUTPUT;
    buf_h[7..0]             : OUTPUT;

    -- Limit Outputs to Motor Controllers
    posxlimit_h             : OUTPUT;
    negxlimit_h             : OUTPUT;
    posylimit_h             : OUTPUT;
    negylimit_h             : OUTPUT;
    poszlimit_h             : OUTPUT;
    negzlimit_h             : OUTPUT;
    poswlimit_h             : OUTPUT;
    negwlimit_h             : OUTPUT;
)

VARIABLE
    clk_r                   : NODE;
    clr_l                   : NODE;

    -- Passthru Transactions
    ptaddr_h[3..0]          : DFF;      -- offset within passthru region
    ptadrinc_h              : NODE;
    ptatn_h                 : LCELL;
    ptburst_h               : LCELL;
    motsel_h                : LCELL;

    -- Pass-Through State Machine
```

```
dqoe2_h                     : LCELL;
dqoe_l                      : LCELL;
ptSM                        : MACHINE OF BITS (
                                 select_h, rd_h, wr_h,
                                 ptadr_h, ptrdy_h, dqoe_h, addonintack_h, motctrlack_h
                              ) WITH STATES (
                                 pt00     = B"00000000",
                                 pt01     = B"10010000",
                                 pt02     = B"10010000",
                                 pt03     = B"11000000",
                                 pt04     = B"11001000",
                                 pt05     = B"10100100",
                                 pt06     = B"10101100",
                                 pt07     = B"00000110",
                                 pt08     = B"10100110",
                                 pt09     = B"00000101",
                                 pt10     = B"00000001"
                              );

-- Mail Box State Machine
mbSM                        : MACHINE OF BITS (
                                 mbena_h, addonint_h
                              ) WITH STATES (
                                 mb00     = B"10",
                                 mb01     = B"00",
                                 mb02     = B"01",
                                 mb03     = B"00",
                                 mb04     = B"10"
                              );
mbff_h[2..0]                : DFFE;
mbcnt_h[3..0]               : DFFE;
mbcnttc_h                   : NODE;

-- Write Control Signals
wrsel_h                     : NODE;
ctrlwr_h                    : LCELL;
mdatwrls_h                  : LCELL;
mdatwrms_h                  : LCELL;
outputwr_h                  : LCELL;
fullmicrowr_h               : LCELL;

-- Control/Status Register
mdat_h[15..00]              : DFFE;
mio_h[2..0]                 : DFFE;
hostintena_h[1..0]          : DFFE;
photosensorff_h[1..0]       : DFF;
photoint_h                  : SRFF;
photointena_h               : DFFE;
fulltravel_h                : DFFE;

-- Output Register
tenxsel_h                   : DFFE;
tenxena_l                   : DFFE;
scansel_h                   : DFFE;
scanena_l                   : DFFE;
spare_l[1..0]               : DFFE;
oc_h[1..0]                  : DFFE;
buf_h[7..0]                 : DFFE;
enablex_l                   : DFFE;
enabley_l                   : DFFE;
enablez_l                   : DFFE;
enablew_l                   : DFFE;
brakex_l                    : DFFE;
brakey_l                    : DFFE;
```

```
    brakez_l                : DFFE;
    brakew_l                : DFFE;
    overridex_h             : DFFE;
    overridey_h             : DFFE;
    overridez_h             : DFFE;
    overridew_h             : DFFE;

    -- Full/Micro Step Limit Registers
    fulllimit_h[7..0]       : DFFE;
    microlimit_h[7..0]      : DFFE;

    -- Read Operations
    dqtri_h[31..00]         : TRI;

    -- Full/Micro Step Selectors
    fullstepx               : fullstep;
    fullstepy               : fullstep;
    fullstepz               : fullstep;
    fullstepw               : fullstep;

    -- Motor Limit Processing
    posydly_h[15..00]       : DFF;
    posydlytc_h             : SRFF;

    -- Motor Controller State Machine
    mcSM                    : MACHINE OF BITS (
                                motctrlwr_h, motctrlrd_h, mdatms_h, mdatls_h,
                                hostslct_h, hostwrite_h, hostread_h, mioclr_h
                            ) WITH STATES (
                                mc00 = B"00000000",
                                mc01 = B"10001000",
                                mc02 = B"01001000",
                                mc03 = B"10101100",
                                mc04 = B"10101100",
                                mc05 = B"10101000",
                                mc06 = B"10101000",
                                mc07 = B"10011100",
                                mc08 = B"10011100",
                                mc09 = B"10011000",
                                mc10 = B"01001010",
                                mc11 = B"01001010",
                                mc12 = B"01101010",
                                mc13 = B"01001000",
                                mc14 = B"01001000",
                                mc15 = B"01001010",
                                mc16 = B"01001010",
                                mc17 = B"01011010",
                                mc18 = B"01001000",
                                mc19 = B"00000001"
                            );

BEGIN
    clk_r   = GLOBAL (bpclk_r);
    clr_l   = GLOBAL (sysrst_l);

    %==================================================================
    Pass Thru Transactions
    ==================================================================%

    -- Passthru Address Register
    ptaddr_h[].clk      = clk_r;
    !ptaddr_h[].clrn    = !clr_l;
    IF ptSM == pt02 THEN
        ptaddr_h[].d = dq_h[05..02];
    ELSIF ptadrinc_h THEN
```

```
    ptaddr_h[].d = ptaddr_h[].q + 1;
ELSE
    ptaddr_h[].d = ptaddr_h[].q;
END IF;
ptadrinc_h = ptrdy_h & ptatn_h;

-- Input Bits
ptatn_h      = !ptatn_l;                -- (LCELL) compensate for clk buf and 0 hld
ptburst_h    = !ptburst_l;              -- (LCELL) compensate for clk buf and 0 hld
motsel_h     = (ptbe_l[] == B"0000")
             & (ptnum_h[] == motorCtrl); -- (LCELL) compensate for clk buf and 0 hld

-- Output Bits
!select_l          = select_h;
!rd_l              = rd_h;
!wr_l              = wr_h;
!ptadr_l           = ptadr_h;
!ptrdy_l           = ptrdy_h;

%=========================================================================
Pass Thru State Machine
=========================================================================%
ptSM.clk     = clk_r;
ptSM.reset   = !clr_l;

-- Due to the relatively slow 12 ns turnoff delay of the AMCC output
-- buffers, we need to add some delay before turning our dq_h[] buffers
-- on. Note that when we turn our buffers off, we want to minimize delay.
dqoe2_h      = dqoe_h;               -- LCELL
!dqoe_l      = dqoe_h & dqoe2_h;     -- LCELL

CASE ptSM IS
    WHEN pt00 =>          -- <nothing> active
        IF ptatn_h THEN
            IF motsel_h THEN ptSM = pt01;
            END IF;
        ELSIF addonint_h THEN ptSM = pt07;
        ELSIF motctrlwr_h THEN ptSM = pt09;
        ELSIF motctrlrd_h THEN ptSM = pt10;
        END IF;

    -- Get Passthru Address
    WHEN pt01 =>          -- select_h, ptadr_h active
        ptSM = pt02;
    WHEN pt02 =>          -- select_h, ptadr_h active
        IF ptwr_h THEN ptSM = pt03;
        ELSE ptSM = pt05;
        END IF;

    -- Passthru Write Operations
    WHEN pt03 =>          -- select_h, rd_h active
        IF !ptburst_h & !ptatn_h THEN ptSM = pt00;
        ELSE ptSM = pt04;
        END IF;
    WHEN pt04 =>          -- select_h, rd_h, ptrdy_h active
        IF ptatn_h THEN ptSM = pt03;
        END IF;

    -- Passthru Read Operations
    WHEN pt05 =>          -- select_h, wr_h, dqoe_h active
        IF !ptburst_h & !ptatn_h THEN ptSM = pt00;
        ELSE ptSM = pt06;
        END IF;
    WHEN pt06 =>          -- select_h, wr_h, ptrdy_h, dqoe_h active
```

```
        IF ptatn_h THEN ptSM = pt05;
        END IF;

    -- Addon Interrupt Operations
    WHEN pt07 =>        -- dqoe_h, addonintack_h active
        ptSM = pt08;
    WHEN pt08 =>        -- select_h, wr_h, dqoe_h, addonintack_h active
        ptSM = pt00;

    -- Motor Controller Write Operations
    WHEN pt09 =>        -- dqoe_h, motctrlack_h active
        IF !motctrlwr_h THEN ptSM = pt00;
        END IF;

    -- Motor Controller Read Operations
    WHEN pt10 =>        -- motctrlack_h active
        IF !motctrlrd_h THEN ptSM = pt00;
        END IF;
END CASE;

%========================================================================
Mail Box State Machine
========================================================================%
mbSM.clk    = clk_r;
mbSM.reset  = !clr_l;

CASE mbSM IS
    -- ==== Interrupts not currently active ====
    WHEN mb00 =>        -- mbena_h active (enables mbcnt_h[] AND mbff_h[])
        mbSM = mb01;
    WHEN mb01 =>        -- <nothing> active
        IF mbff_h[].q != B"000" THEN mbSM = mb02;
        ELSE mbSM = mb00;
        END IF;

    -- ==== Interrupts currently active ====
    -- Generate an interrupt
    WHEN mb02 =>        -- addonint_h active
        IF addonintack_h THEN mbSM = mb03;
        END IF;

    -- Loop awhile, checking to see if interrupts are acknowledged
    WHEN mb03 =>        -- <nothing> active
        IF mbff_h[].q == B"000" THEN mbSM = mb00;
        ELSE mbSM = mb04;
        END IF;
    WHEN mb04 =>        -- mbena_h active
        IF mbcnttc_h THEN mbSM = mb02;
        ELSE mbSM = mb03;
        END IF;
END CASE;

mbff_h[].clk    = clk_r;
!mbff_h[].clrn  = !clr_l;
mbff_h[].ena    = mbena_h;
mbff_h[2].d     = photointena_h & photoint_h;
mbff_h[1].d     = hostintena_h[1] & !hostint_l[1];
mbff_h[0].d     = hostintena_h[0] & !hostint_l[0];

mbcnt_h[].clk   = clk_r;
!mbcnt_h[].clrn = !clr_l;
mbcnt_h[].ena   = mbena_h;
IF (mbSM == mb00) THEN
    mbcnt_h[].d = 0;
```

```
ELSE
    mbcnt_h[].d = mbcnt_h[].q + 1;
END IF;
mbcnttc_h       = (mbcnt_h[].q == H"F");

%=============================================================================
Write Operations
=============================================================================%
wrsel_h             = rd_h & ptrdy_h & ptatn_h;      -- "rd_h & ptrdy_h" is same as 'pt04'!
ctrlwr_h            = wrsel_h & (ptaddr_h[] == motCtrlStat);     -- LCELL
mdatwrls_h          = wrsel_h & (ptaddr_h[] == motCtrlStat)
                        # motctrlrd_h & mdatls_h;                -- LCELL
mdatwrms_h          = wrsel_h & (ptaddr_h[] == motCtrlStat)
                        # motctrlrd_h & mdatms_h;                -- LCELL
outputwr_h          = wrsel_h & (ptaddr_h[] == motOutput);       -- LCELL
fullmicrowr_h       = wrsel_h & (ptaddr_h[] == motFullMicro);    -- LCELL

-- Control/Status Register Bits
mdat_h[].clk        = clk_r;
!mdat_h[].clrn      = !clr_l;
mdat_h[15..08].ena  = mdatwrms_h;
mdat_h[07..00].ena  = mdatwrls_h;
IF mdatls_h THEN
    mdat_h[07..00].d = dq_h[07..00];
ELSIF mdatms_h THEN
    mdat_h[15..08].d = dq_h[07..00];
ELSE
    mdat_h[15..00].d = dq_h[15..00];
END IF;

-- mio_h[] values:
--   B"111" Data Write from chip set #1
--   B"110" Data Read from chip set #1
--   B"101" Command Write to chip set #1
--   B"100" No operation (will reset to B"000")
--   B"011" Data Write to chip set #0
--   B"010" Data Read from chip set #0
--   B"001" Command Write to chip set #0
--   B"000" No operation
mio_h[].clk         = clk_r;
!mio_h[].clrn       = !sysrst_l # mioclr_h;
mio_h[].ena         = ctrlwr_h;
mio_h[].d           = dq_h[18..16];

hostintena_h[].clk      = clk_r;
!hostintena_h[].clrn    = !clr_l;
hostintena_h[].ena      = ctrlwr_h;
hostintena_h[].d        = dq_h[24..23];

-- 2 bit shift register used as a rising edge detector of photosensor_h
photosensorff_h[].clk   = clk_r;
!photosensorff_h[].clrn = !clr_l;
photosensorff_h[0].d    = photosensor_h;
photosensorff_h[1].d    = photosensorff_h[0].q;

photoint_h.clk      = clk_r;
!photoint_h.clrn    = !clr_l;
photoint_h.r        = photoint_h.q & ctrlwr_h & !dq_h[25];
photoint_h.s        = !photoint_h.q & photosensorff_h[0] & !photosensorff_h[1];

photointena_h.clk       = clk_r;
!photointena_h.clrn     = !clr_l;
photointena_h.ena       = ctrlwr_h;
photointena_h.d         = dq_h[26];
```

```
fulltravel_h.clk        = clk_r;
!fulltravel_h.clrn      = !clr_l;
fulltravel_h.ena        = ctrlwr_h;
fulltravel_h.d          = dq_h[27];

-- Output Register Bits
tenxsel_h.clk       = clk_r;
!tenxsel_h.clrn     = !clr_l;
tenxsel_h.ena       = outputwr_h;
tenxsel_h.d         = dq_h[00];

tenxena_l.clk       = clk_r;
!tenxena_l.prn      = !clr_l;
tenxena_l.ena       = outputwr_h;
!tenxena_l.d        = dq_h[01];

scansel_h.clk       = clk_r;
!scansel_h.clrn     = !clr_l;
scansel_h.ena       = outputwr_h;
scansel_h.d         = dq_h[02];

scanena_l.clk       = clk_r;
!scanena_l.prn      = !clr_l;
scanena_l.ena       = outputwr_h;
!scanena_l.d        = dq_h[03];

spare_l[].clk       = clk_r;
!spare_l[].prn      = !clr_l;
spare_l[].ena       = outputwr_h;
!spare_l[].d        = dq_h[05..04];

oc_h[].clk          = clk_r;
!oc_h[].clrn        = !clr_l;
oc_h[].ena          = outputwr_h;
oc_h[].d            = dq_h[07..06];

buf_h[].clk         = clk_r;
!buf_h[].clrn       = !clr_l;
buf_h[].ena         = outputwr_h;
buf_h[].d           = dq_h[15..08];

enablex_l.clk       = clk_r;
!enablex_l.prn      = !clr_l;
enablex_l.ena       = outputwr_h;
!enablex_l.d        = dq_h[16];

enabley_l.clk       = clk_r;
!enabley_l.prn      = !clr_l;
enabley_l.ena       = outputwr_h;
!enabley_l.d        = dq_h[17];

enablez_l.clk       = clk_r;
!enablez_l.prn      = !clr_l;
enablez_l.ena       = outputwr_h;
!enablez_l.d        = dq_h[18];

enablew_l.clk       = clk_r;
!enablew_l.prn      = !clr_l;
enablew_l.ena       = outputwr_h;
!enablew_l.d        = dq_h[19];

brakex_l.clk        = clk_r;
!brakex_l.prn       = !clr_l;
```

```
brakex_l.ena         = outputwr_h;
!brakex_l.d          = dq_h[20];

brakey_l.clk         = clk_r;
!brakey_l.prn        = !clr_l;
brakey_l.ena         = outputwr_h;
!brakey_l.d          = dq_h[21];

brakez_l.clk         = clk_r;
!brakez_l.prn        = !clr_l;
brakez_l.ena         = outputwr_h;
!brakez_l.d          = dq_h[22];

brakew_l.clk         = clk_r;
!brakew_l.prn        = !clr_l;
brakew_l.ena         = outputwr_h;
!brakew_l.d          = dq_h[23];

overridex_h.clk      = clk_r;
!overridex_h.clrn    = !clr_l;
overridex_h.ena      = outputwr_h;
overridex_h.d        = dq_h[24];

overridey_h.clk      = clk_r;
!overridey_h.clrn    = !clr_l;
overridey_h.ena      = outputwr_h;
overridey_h.d        = dq_h[25];

overridez_h.clk      = clk_r;
!overridez_h.clrn    = !clr_l;
overridez_h.ena      = outputwr_h;
overridez_h.d        = dq_h[26];

overridew_h.clk      = clk_r;
!overridew_h.clrn    = !clr_l;
overridew_h.ena      = outputwr_h;
overridew_h.d        = dq_h[27];

-- Full/Micro Step Register
fulllimit_h[].clk        = clk_r;
!fulllimit_h[].clrn      = !clr_l;
fulllimit_h[].ena        = fullmicrowr_h;
fulllimit_h[].d          = dq_h[29..22];     -- smaller number (higher rpm limit)
-- fulllimit_h[].d       = dq_h[23..16];     -- simulation

microlimit_h[].clk       = clk_r;
!microlimit_h[].clrn     = !clr_l;
microlimit_h[].ena       = fullmicrowr_h;
microlimit_h[].d         = dq_h[13..06];     -- big number (lower rpm limit)
-- microlimit_h[].d      = dq_h[07..00];     -- simulation

%=====================================================================
Read Operations
=====================================================================%
IF addonintack_h THEN
    dqtri_h[31..03] = 0;
    dqtri_h[02..00] = mbff_h[];      -- Mailbox #4, byte #1 (#3 is unavailable)
ELSIF mdatms_h THEN
    dqtri_h[31..08] = GND;
    dqtri_h[07..00] = mdat_h[15..08];
ELSIF mdatls_h THEN
    dqtri_h[31..08] = GND;
    dqtri_h[07..00] = mdat_h[07..00];
ELSE
```

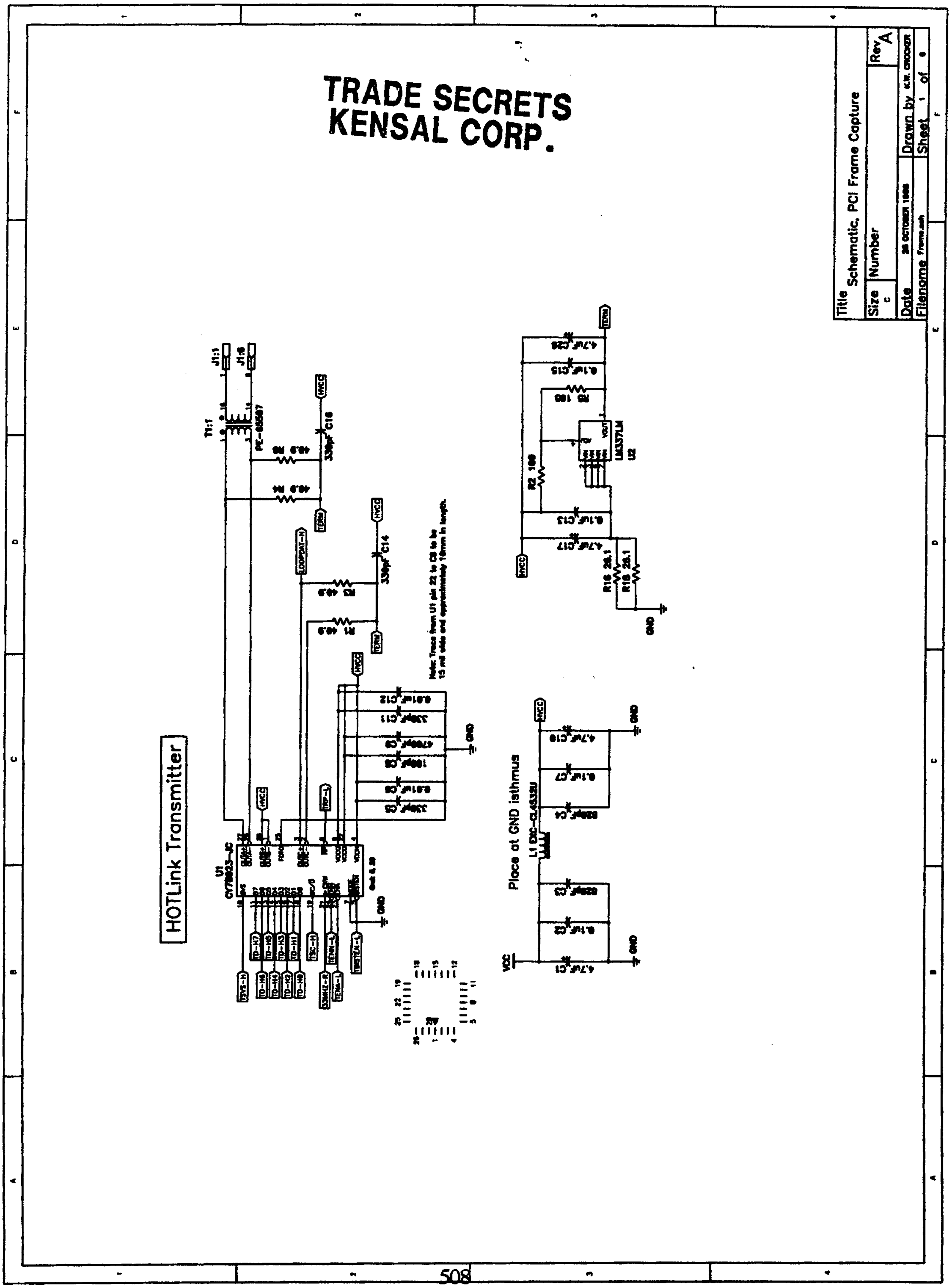
```
CASE ptaddr_h[] IS
    WHEN motCtrlStat =>
        dqtri_h[31..28] = GND;
        dqtri_h[27]     = fulltravel_h;
        dqtri_h[26]     = photointena_h;
        dqtri_h[25]     = photoint_h;
        dqtri_h[24..23] = hostintena_h[1..0];
        dqtri_h[22..21] = !hostint_l[1..0];
        dqtri_h[20..19] = hostrdy_h[1..0];
        dqtri_h[18..16] = mio_h[2..0];
        dqtri_h[15..00] = mdat_h[15..00];
    WHEN motLimit =>
        dqtri_h[31..24] = GND;
        dqtri_h[23]     = negwlimit_h;
        dqtri_h[22]     = poswlimit_h;
        dqtri_h[21]     = negzlimit_h;
        dqtri_h[20]     = poszlimit_h;
        dqtri_h[19]     = negylimit_h;
        dqtri_h[18]     = posylimit_h;
        dqtri_h[17]     = negxlimit_h;
        dqtri_h[16]     = posxlimit_h;
        dqtri_h[15]     = scandislimit_h;
        dqtri_h[14]     = scanenalimit_h;
        dqtri_h[13]     = fortyxlimit_h;
        dqtri_h[12]     = tenxlimit_h;
        dqtri_h[11]     = sparelimit_h3;
        dqtri_h[10]     = photosensor_h;
        dqtri_h[09]     = nzlimit_h;
        dqtri_h[08]     = pzlimit_h;
        dqtri_h[07..06] = sparelimit_h[2..1];
        dqtri_h[05]     = nylimit_h;
        dqtri_h[04]     = pylimit_h;
        dqtri_h[03]     = sparelimit_h[0];
        dqtri_h[02]     = ldlimit_h;
        dqtri_h[01]     = nxlimit_h;
        dqtri_h[00]     = pxlimit_h;
    WHEN motOutput =>
        dqtri_h[31..28] = GND;
        dqtri_h[27]     = overridew_h;
        dqtri_h[26]     = overridez_h;
        dqtri_h[25]     = overridey_h;
        dqtri_h[24]     = overridex_h;
        dqtri_h[23]     = !brakew_l;
        dqtri_h[22]     = !brakez_l;
        dqtri_h[21]     = !brakey_l;
        dqtri_h[20]     = !brakex_l;
        dqtri_h[19]     = !enablew_l;
        dqtri_h[18]     = !enablez_l;
        dqtri_h[17]     = !enabley_l;
        dqtri_h[16]     = !enablex_l;
        dqtri_h[15..08] = buf_h[7..0];
        dqtri_h[07..06] = oc_h[1..0];
        dqtri_h[05..04] = !spare_l[1..0];
        dqtri_h[03]     = !scanena_l;
        dqtri_h[02]     = scansel_h;
        dqtri_h[01]     = !tenxena_l;
        dqtri_h[00]     = tenxsel_h;
    WHEN motFullMicro =>
        dqtri_h[31..30] = GND;
        dqtri_h[29..22] = fulllimit_h[7..0];
        dqtri_h[21..14] = GND;
        dqtri_h[13..06] = microlimit_h[7..0];
        dqtri_h[05..00] = GND;
END CASE;
```

```
END IF;

dqtri_h[].oe      = !dqoe_l;
dq_h[]            = dqtri_h[].out;

%===========================================================================
'adr_h[]' Multiplexer
===========================================================================%
IF addonintack_h THEN
    adr_h[] = AOMB4;
ELSE
    adr_h[] = APTD;
END IF;

%===========================================================================
Full/Micro Step Operations, Braking and Mode Control
===========================================================================%
fullstepx.(clk_r, clr_l, fulllimit_h[7..0], microlimit_h[7..0], override_h, brake_l, phase_h[1..0])
    = (clk_r, clr_l, fulllimit_h[7..0], microlimit_h[7..0], overridex_h, brakex_l, phasex_h[1..0]);
(modex_h[1..0], selfullx_h[1..0]) = fullstepx.(mode_h[1..0], selfull_h[1..0]);

fullstepy.(clk_r, clr_l, fulllimit_h[7..0], microlimit_h[7..0], override_h, brake_l, phase_h[1..0])
    = (clk_r, clr_l, fulllimit_h[7..0], microlimit_h[7..0], overridey_h, brakey_l, phasey_h[1..0]);
(modey_h[1..0], selfully_h[1..0]) = fullstepy.(mode_h[1..0], selfull_h[1..0]);

fullstepz.(clk_r, clr_l, fulllimit_h[7..0], microlimit_h[7..0], override_h, brake_l, phase_h[1..0])
    = (clk_r, clr_l, fulllimit_h[7..0], microlimit_h[7..0], overridez_h, brakez_l, phasez_h[1..0]);
(modez_h[1..0], selfullz_h[1..0]) = fullstepz.(mode_h[1..0], selfull_h[1..0]);

fullstepw.(clk_r, clr_l, fulllimit_h[7..0], microlimit_h[7..0], override_h, brake_l, phase_h[1..0])
    = (clk_r, clr_l, fulllimit_h[7..0], microlimit_h[7..0], overridew_h, brakew_l, phasew_h[1..0]);
(modew_h[1..0], selfullw_h[1..0]) = fullstepw.(mode_h[1..0], selfull_h[1..0]);

%===========================================================================
Motor Limit Processing
===========================================================================%
-- This delay counter allows some maneuvering room when coming out
-- of negative y limit while in the load channel. For a 33MHz clock
-- we allow 16-bits or 1.966ms.
posydly_h[].clk      = clk_r;
!posydly_h[].clrn    = nylimit_h;
posydly_h[].d        = posydly_h[].q + 1;

posydlytc_h.clk      = clk_r;
!posydlytc_h.clrn    = !clr_l;
posydlytc_h.s        = (posydly_h[] == H"FFFF");
posydlytc_h.r        = nylimit_h;

posxlimit_h = pxlimit_h;
negxlimit_h = nxlimit_h # (!nylimit_h & ldlimit_h);

posylimit_h = fulltravel_h & pylimit_h
            # !fulltravel_h & posydlytc_h & !nylimit_h;
negylimit_h = nylimit_h;

poszlimit_h = pzlimit_h;
negzlimit_h = nzlimit_h;

-- Setup placeholders for connecting sparelimit_h[] to W axis
poswlimit_h = sparelimit_h[3] & sparelimit_h[2] & sparelimit_h[1] & sparelimit_h[0];
negwlimit_h = sparelimit_h[3] & sparelimit_h[2] & sparelimit_h[1] & sparelimit_h[0];

%===========================================================================
Motor Controller State Machine
```

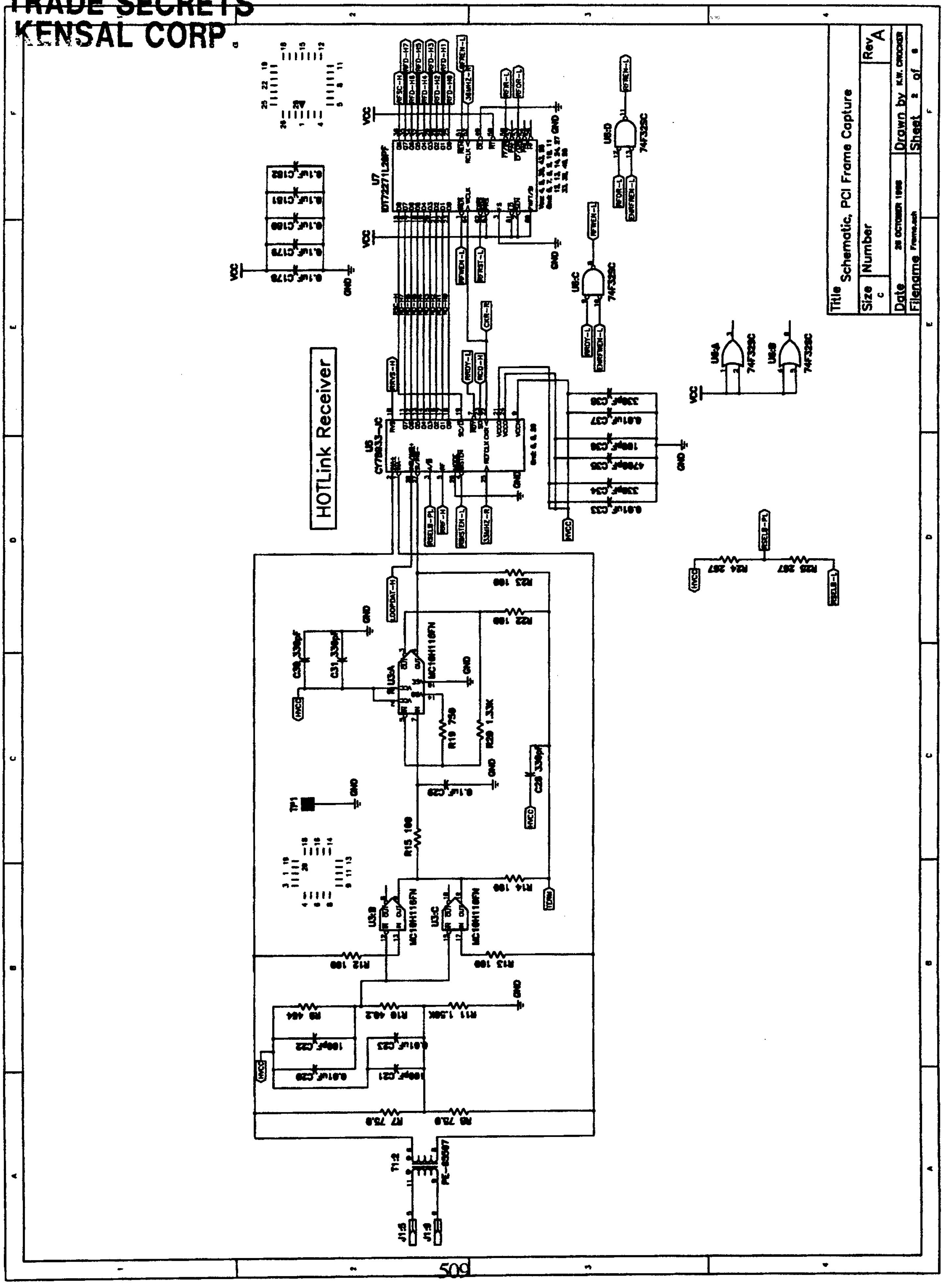
```
=========================================================================%
mcSM.clk    = clk_r;
mcSM.reset  = !clr_l;
CASE mcSM IS
    WHEN mc00 =>          -- <nothing> active
        IF mio_h[0] THEN mcSM = mc01;          -- command or data write
        ELSIF mio_h[1] THEN mcSM = mc02;       -- data read
        ELSIF mio_h[2] THEN mcSM = mc19;       -- mio_h[] = B"100"... Chipset #1 NOP?!
        END IF;

    -- Motor Controller Write Operation
    WHEN mc01 =>          -- motctrlwr_h, hostslct_h active
        IF motctrlack_h THEN
            IF mio_h[1] THEN mcSM = mc03;      -- data write operation
            ELSE mcSM = mc07;                  -- command write operation
            END IF;
        END IF;

    -- Motor Controller Read Operation
    WHEN mc02 =>          -- motctrlrd_h, hostslct_h active
        IF motctrlack_h THEN mcSM = mc10;
        END IF;

    -- Data Write (MSB)
    WHEN mc03 =>          -- motctrlwr_h, mdatms_h, hostslct_h, hostwrite_h active
        mcSM = mc04;
    WHEN mc04 =>          -- motctrlwr_h, mdatms_h, hostslct_h, hostwrite_h active
        mcSM = mc05;
    WHEN mc05 =>          -- motctrlwr_h, mdatms_h, hostslct_h active
        mcSM = mc06;
    WHEN mc06 =>          -- motctrlwr_h, mdatms_h, hostslct_h active
        mcSM = mc07;

    -- Command Byte Write (or Data Write LSB)
    WHEN mc07 =>          -- motctrlwr_h, mdatls_h, hostslct_h, hostwrite_h active
        mcSM = mc08;
    WHEN mc08 =>          -- motctrlwr_h, mdatls_h, hostslct_h, hostwrite_h active
        mcSM = mc09;
    WHEN mc09 =>          -- motctrlwr_h, mdatls_h, hostslct_h active
        mcSM = mc19;

    -- Data Read
    WHEN mc10 =>          -- motctrlrd_h, hostslct_h, hostread_h active
        mcSM = mc11;
    WHEN mc11 =>          -- motctrlrd_h, hostslct_h, hostread_h active
        mcSM = mc12;
    WHEN mc12 =>          -- motctrlrd_h, mdatms_h, hostslct_h, hostread_h active
        mcSM = mc13;
    WHEN mc13 =>          -- motctrlrd_h, hostslct_h active
        mcSM = mc14;
    WHEN mc14 =>          -- motctrlrd_h, hostslct_h active
        mcSM = mc15;
    WHEN mc15 =>          -- motctrlrd_h, hostslct_h, hostread_h active
        mcSM = mc16;
    WHEN mc16 =>          -- motctrlrd_h, hostslct_h, hostread_h active
        mcSM = mc17;
    WHEN mc17 =>          -- motctrlrd_h, mdatls_h, hostslct_h, hostread_h active
        mcSM = mc18;
    WHEN mc18 =>          -- motctrlrd_h, hostslct_h active
        mcSM = mc19;

    -- Clear mio_h[] register
    WHEN mc19 =>          -- mioclr_h active
        mcSM = mc00;
```

```
    END CASE;

    IF mio_h[2] THEN           -- Chip set #1 selected
        !hostslct_l[1]  = hostslct_h;
         hostcmd_h[1]   = hostslct_h & !mio_h[1];
        !hostwrite_l[1] = hostwrite_h;
        !hostread_l[1]  = hostread_h;
        !hostslct_l[0]  = GND;
         hostcmd_h[0]   = GND;
        !hostwrite_l[0] = GND;
        !hostread_l[0]  = GND;
    ELSE                       -- Chip set #0 selected
        !hostslct_l[1]  = GND;
         hostcmd_h[1]   = GND;
        !hostwrite_l[1] = GND;
        !hostread_l[1]  = GND;
        !hostslct_l[0]  = hostslct_h;
         hostcmd_h[0]   = hostslct_h & !mio_h[1];
        !hostwrite_l[0] = hostwrite_h;
        !hostread_l[0]  = hostread_h;
    END IF;

END;
```

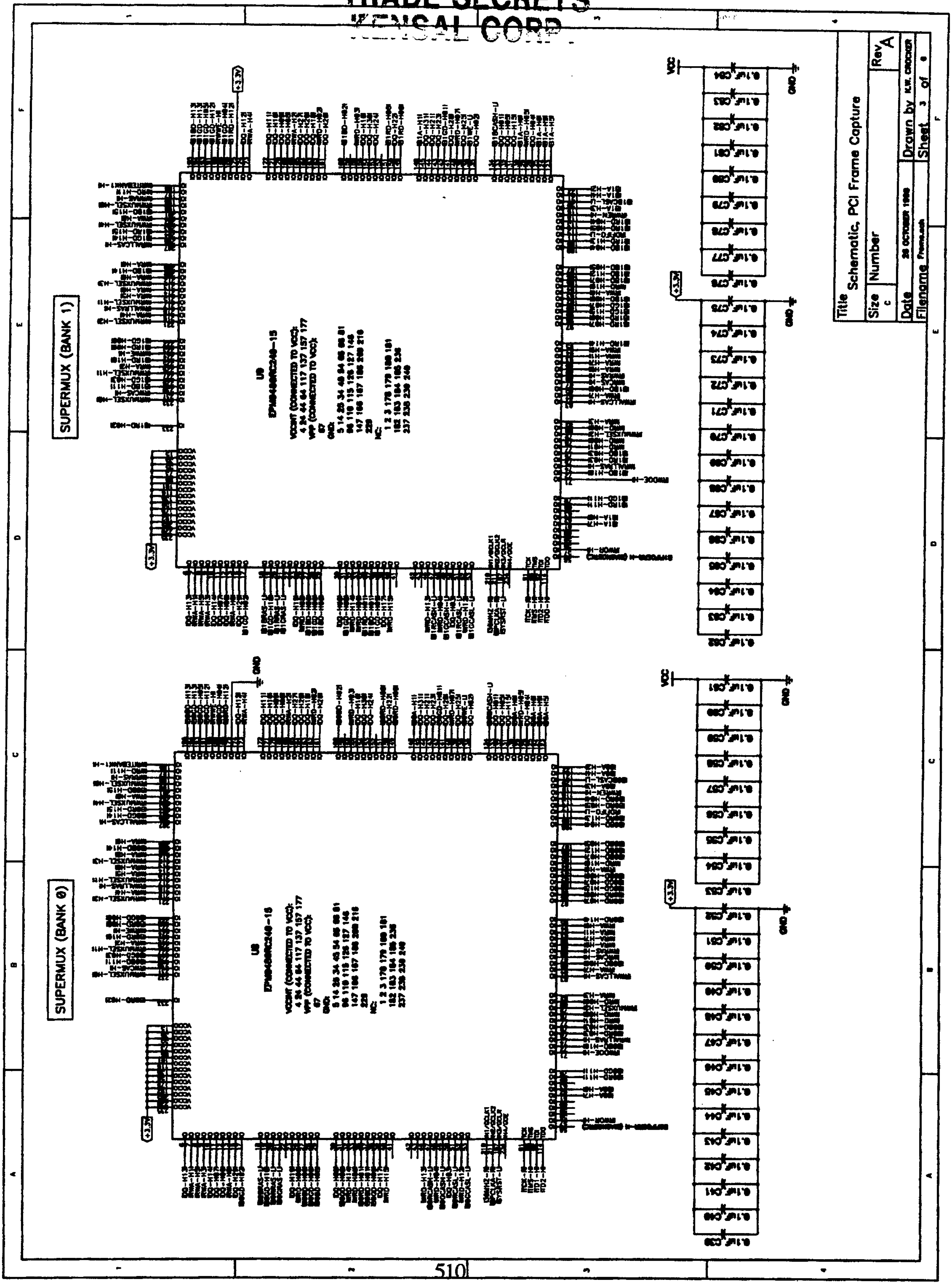
TRADE SECRETS
KENSAL CORP.

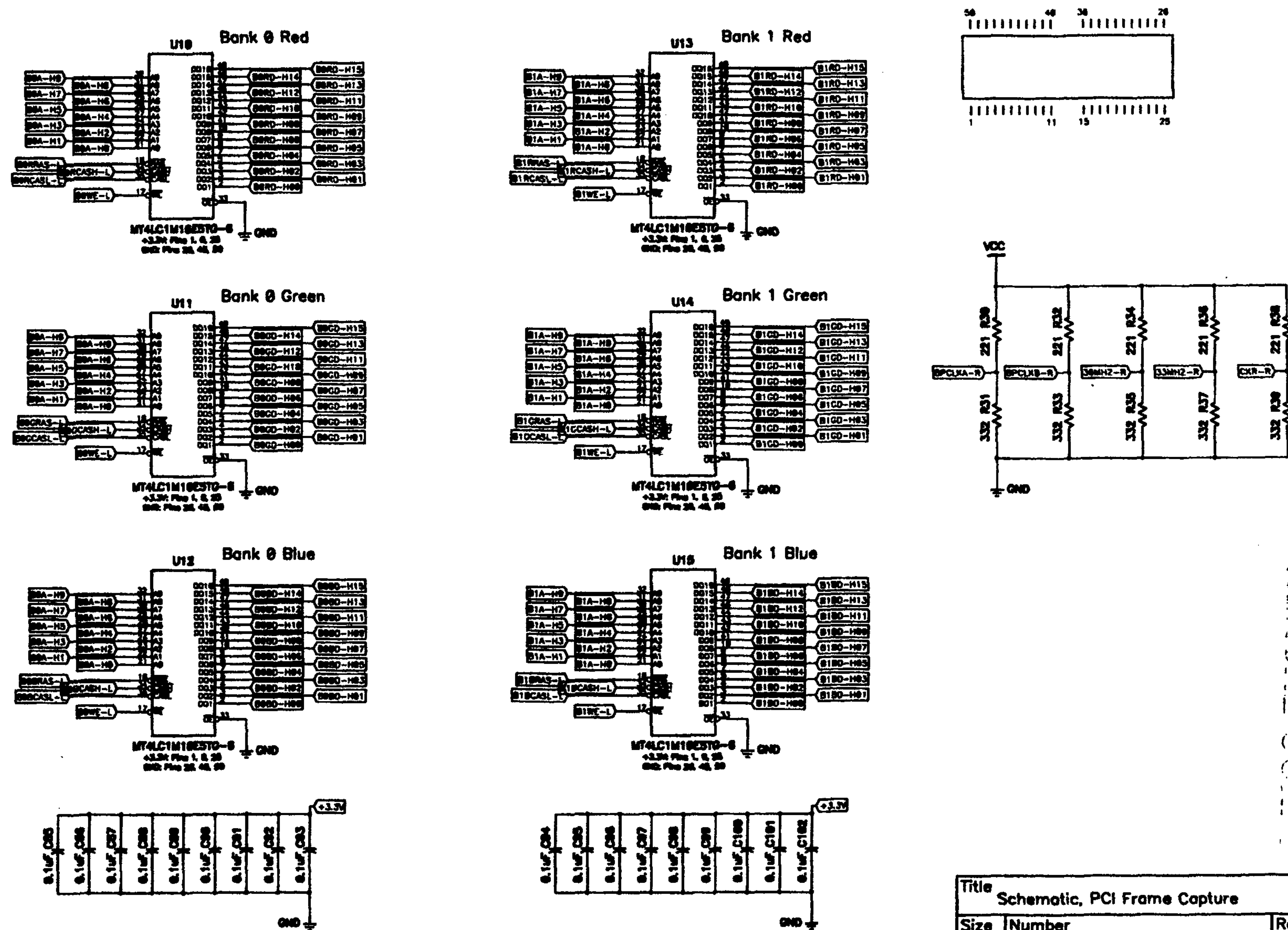


TRADE SECRETS
KENSAL CORP



TRADE SECRETS
KENSAL CORP.

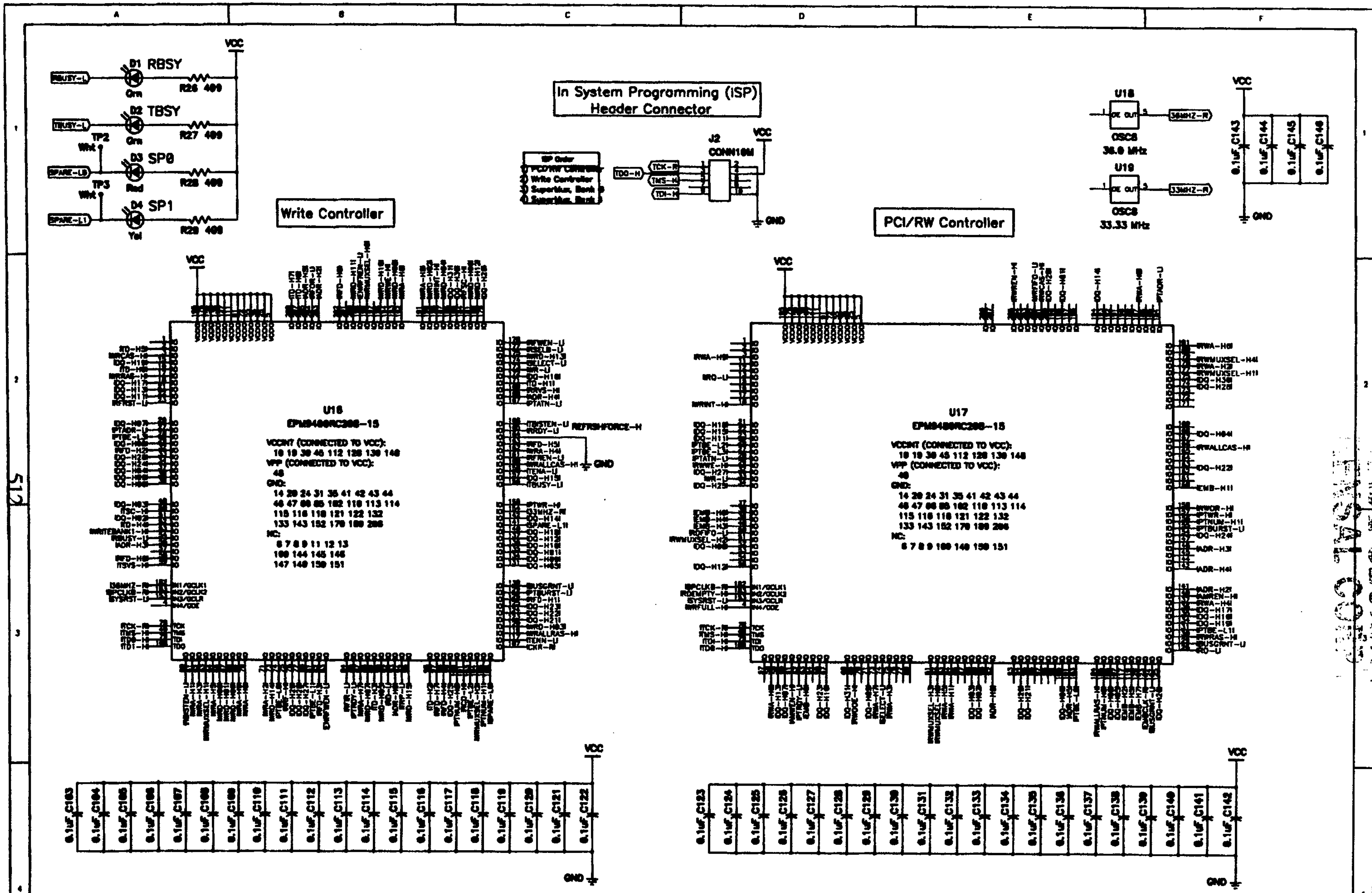




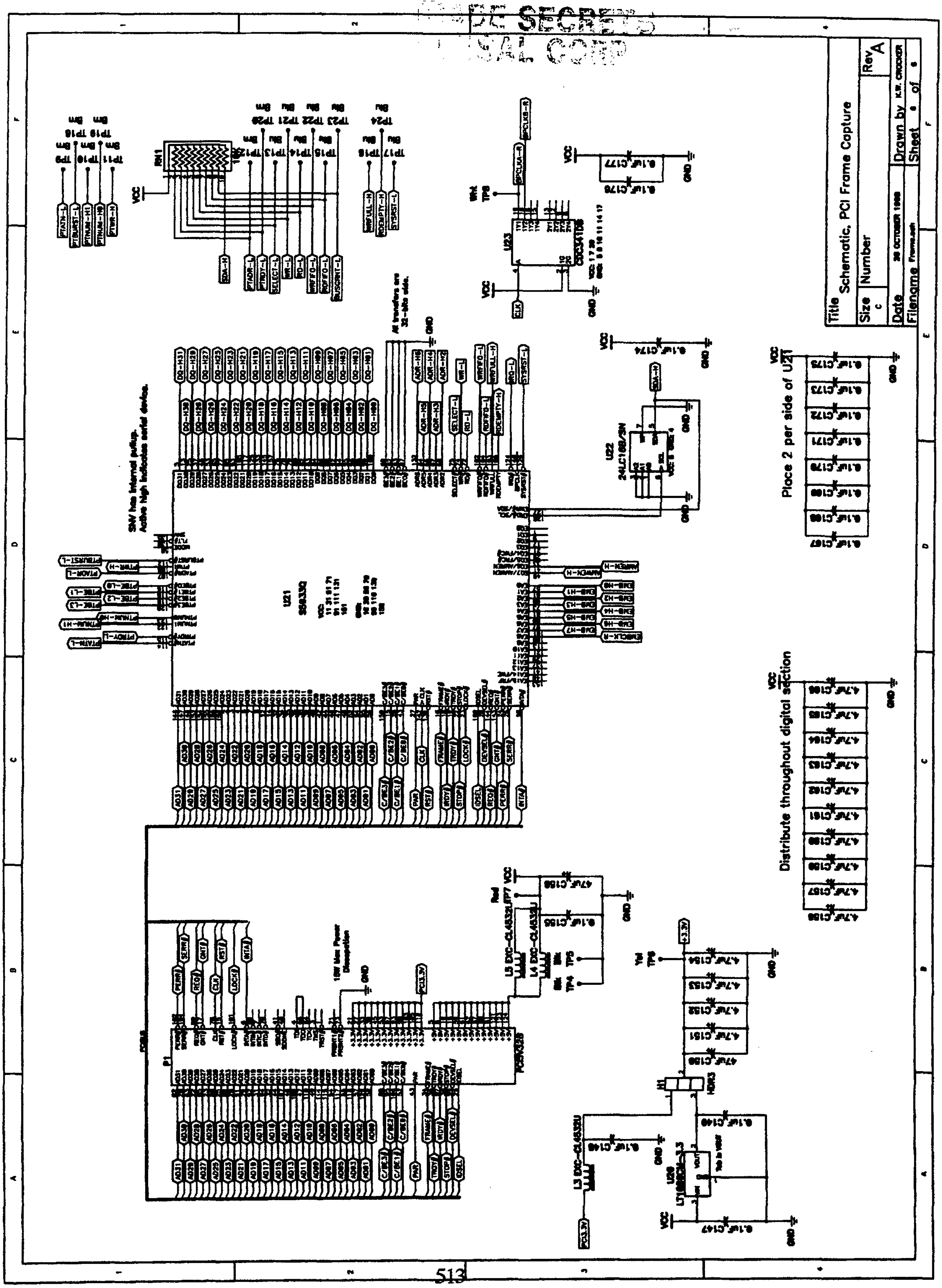
TRADE SECRETS

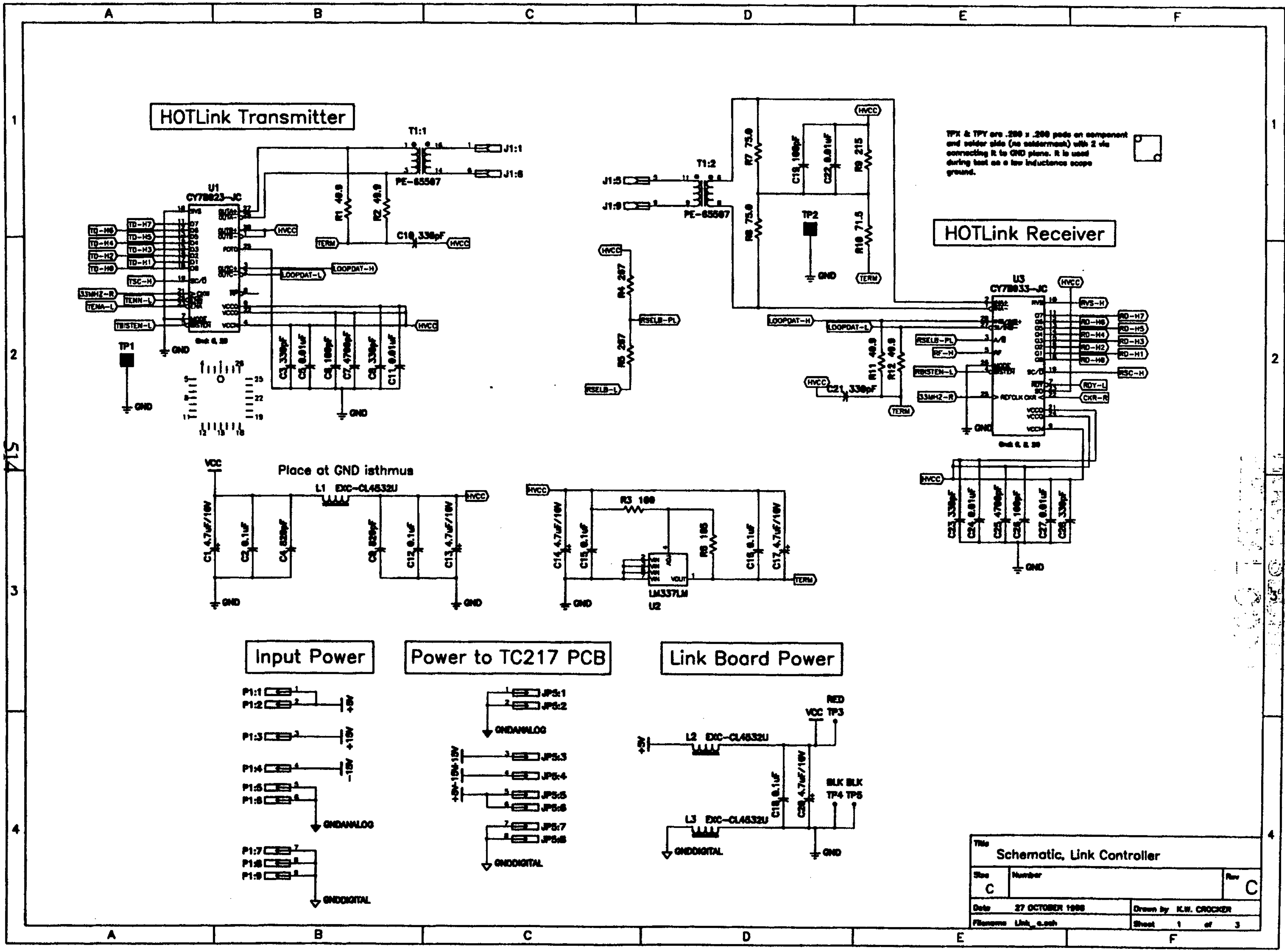
Title Schematic, PCI Frame Capture

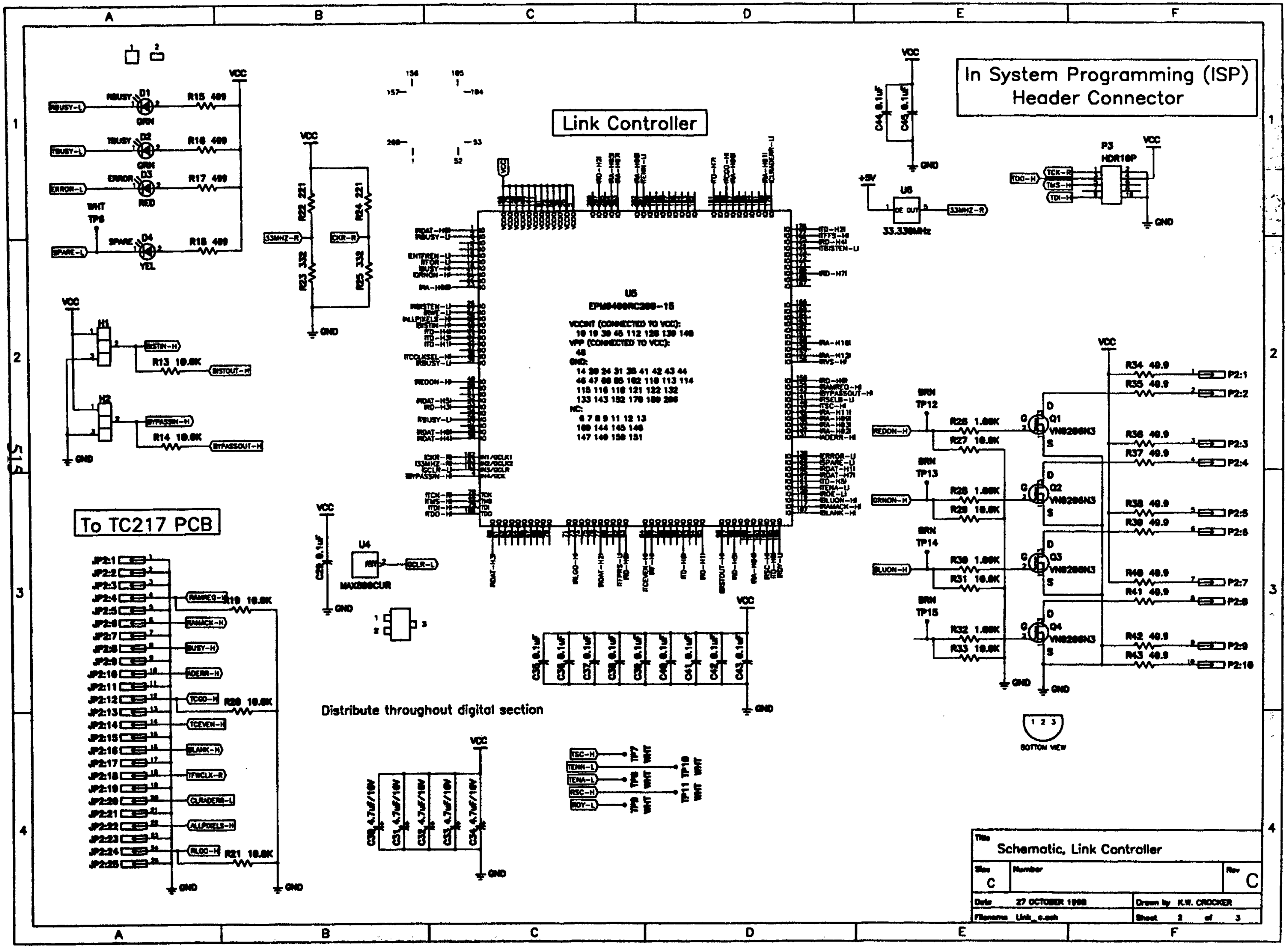
Size C	Number	Rev A
Date 30 OCTOBER 1998		Drawn by K.W. CROCKER
Filename Frame.sch		Sheet 4 of 6



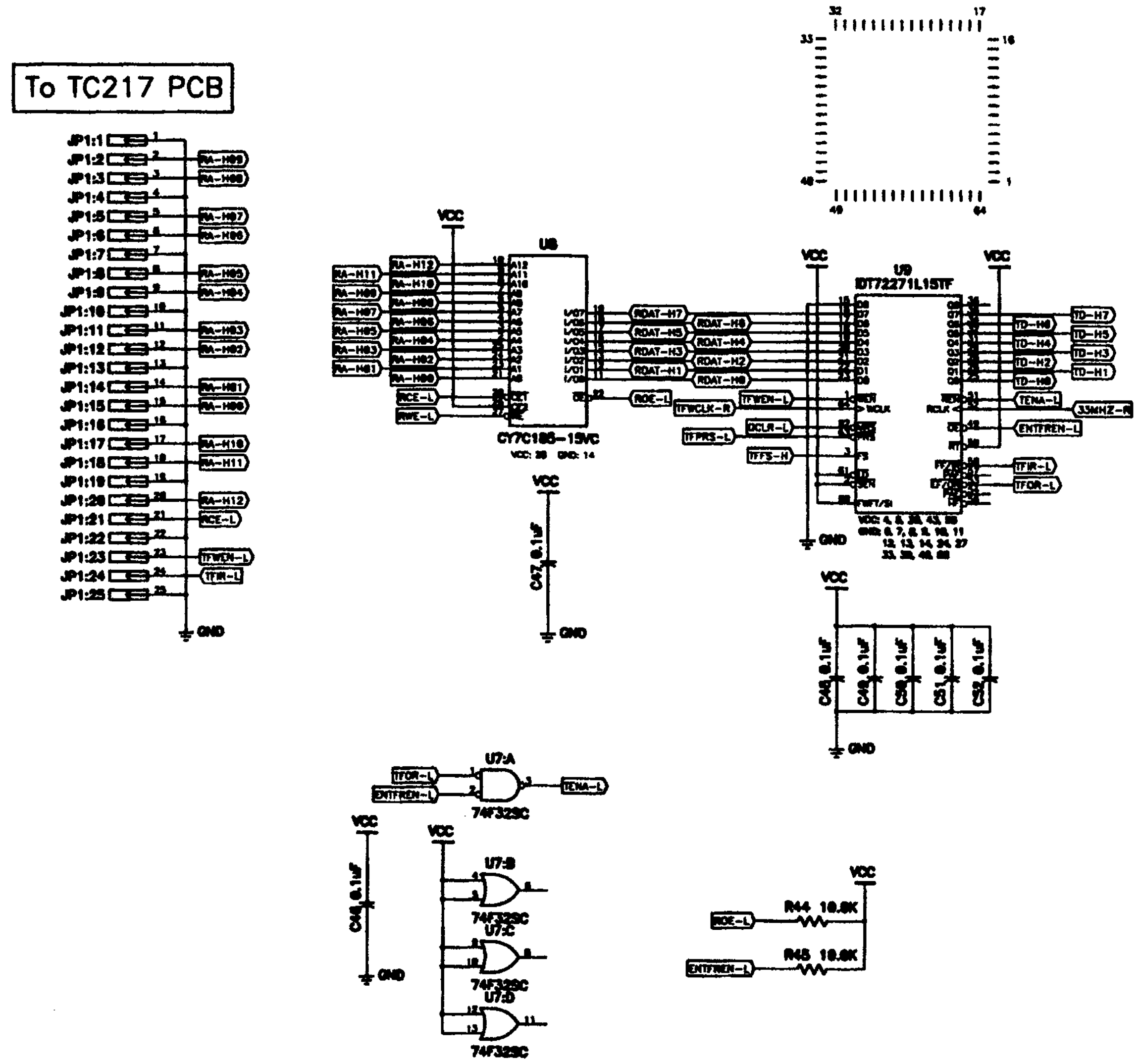
Title: Schematic, PCI Frame Capture			
Size: C	Number		Rev: A
Date: 28 OCTOBER 1998		Drawn by: K.W. CROCKER	
Filename: Frame.sch		Sheet 5 of 6	


Title Schematic, PCI Frame Capture
Size C
Number
Rev A
Date 20 OCTOBER 1998
Filename
Drawn by K.W. CROCKER
Sheet 6 of 6
Place 2 per side of U21
Distribute throughout digital section
SW has internal pullup.
Active high indicates serial device.
All transfers are 32-bit wide.
15W Max Power Dissipation
U21 S5933Q
U22 24LC16B/SN
U23 CDC341DB
P1 PCI5V32B
H1 HDR3
U26 LT1086CM-3.3
RN1
VCC
GND


HOTLink Transmitter
HOTLink Receiver
Place at GND isthmus
Input Power
Power to TC217 PCB
Link Board Power
U1 CY7B923-JC
U3 CY7B933-JC
T1:1 PE-65507
T1:2 PE-65507
U2 LM337LM
L1 EXC-CL4532U
L2 EXC-CL4532U
L3 EXC-CL4532U
GNDANALOG
GNDDIGITAL
Schematic, Link Controller
Size C
Rev C
Date 27 OCTOBER 1998
Drawn by K.W. CROCKER
Filename Link_c.sch
Sheet 1 of 3


In System Programming (ISP) Header Connector
Link Controller
To TC217 PCB
Distribute throughout digital section
U5
EPM9400RC208-15
VCCINT (CONNECTED TO VCC):
10 19 30 45 112 128 139 148
VPP (CONNECTED TO VCC):
48
GND:
14 20 24 31 38 41 42 43 44
46 47 86 85 102 110 113 114
115 116 119 121 122 132
133 143 152 170 180 206
NC:
6 7 8 9 11 12 13
109 144 145 146
147 149 150 151
U6
33.330MHz
U4
MAX809CUR
P3
HDR10P
BOTTOM VIEW
Title
Schematic, Link Controller
Size C
Rev C
Date 27 OCTOBER 1998
Drawn by K.W. CROCKER
Filename Link_c.sch
Sheet 2 of 3

KENSAL CORP. TRADE SECRETS

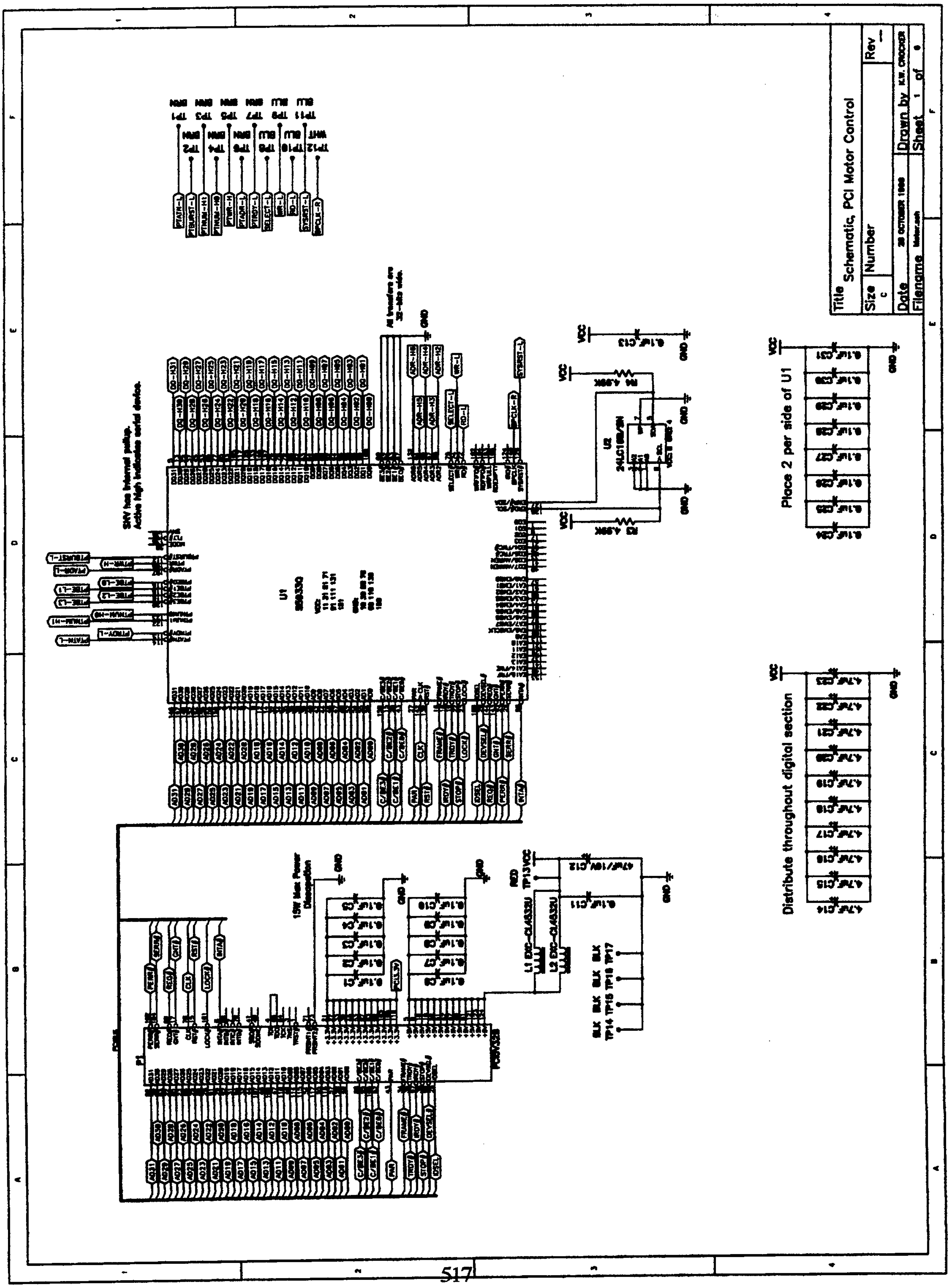


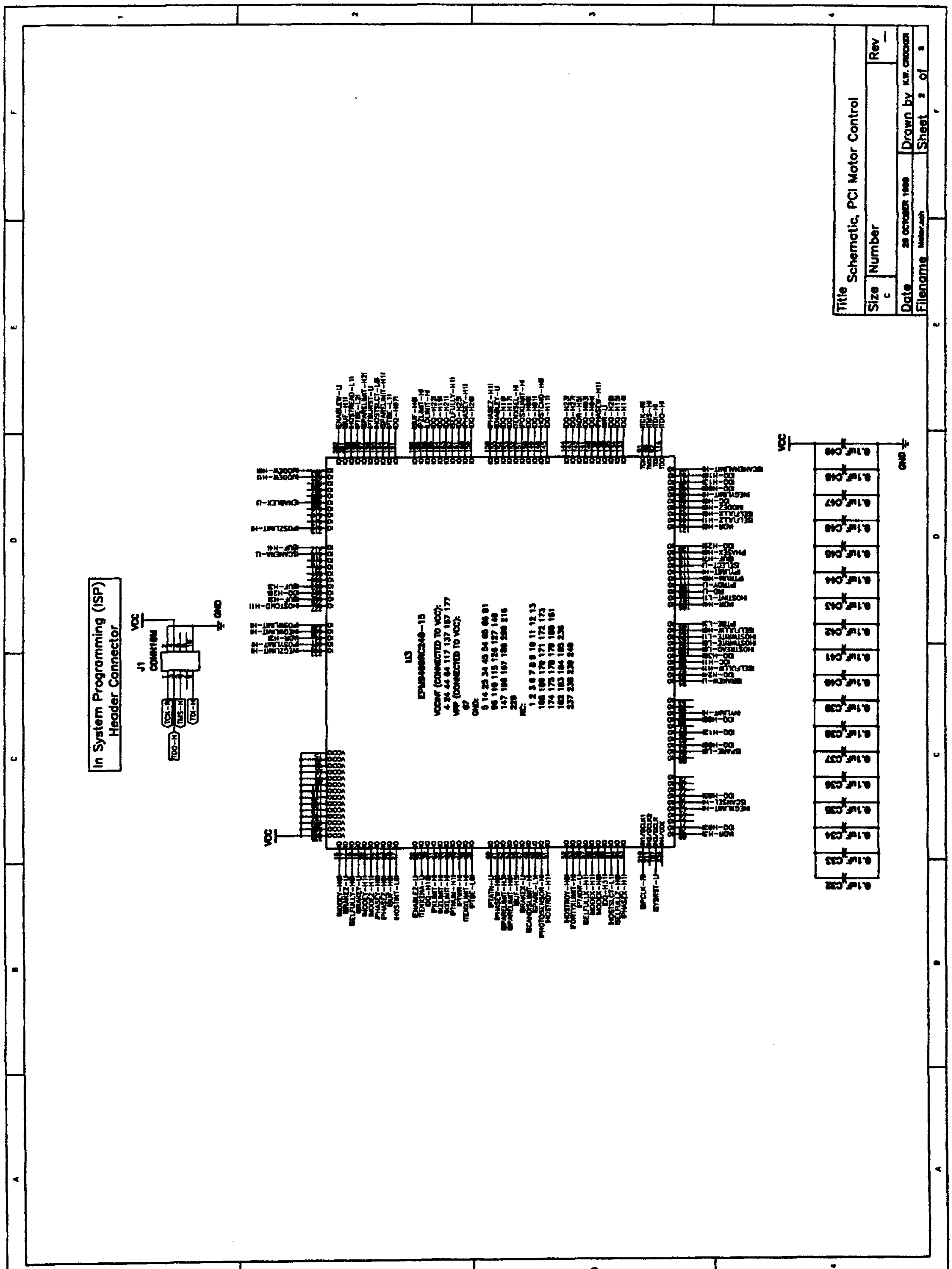
Title: Schematic, Link Controller

Size	Number	Rev
C		C

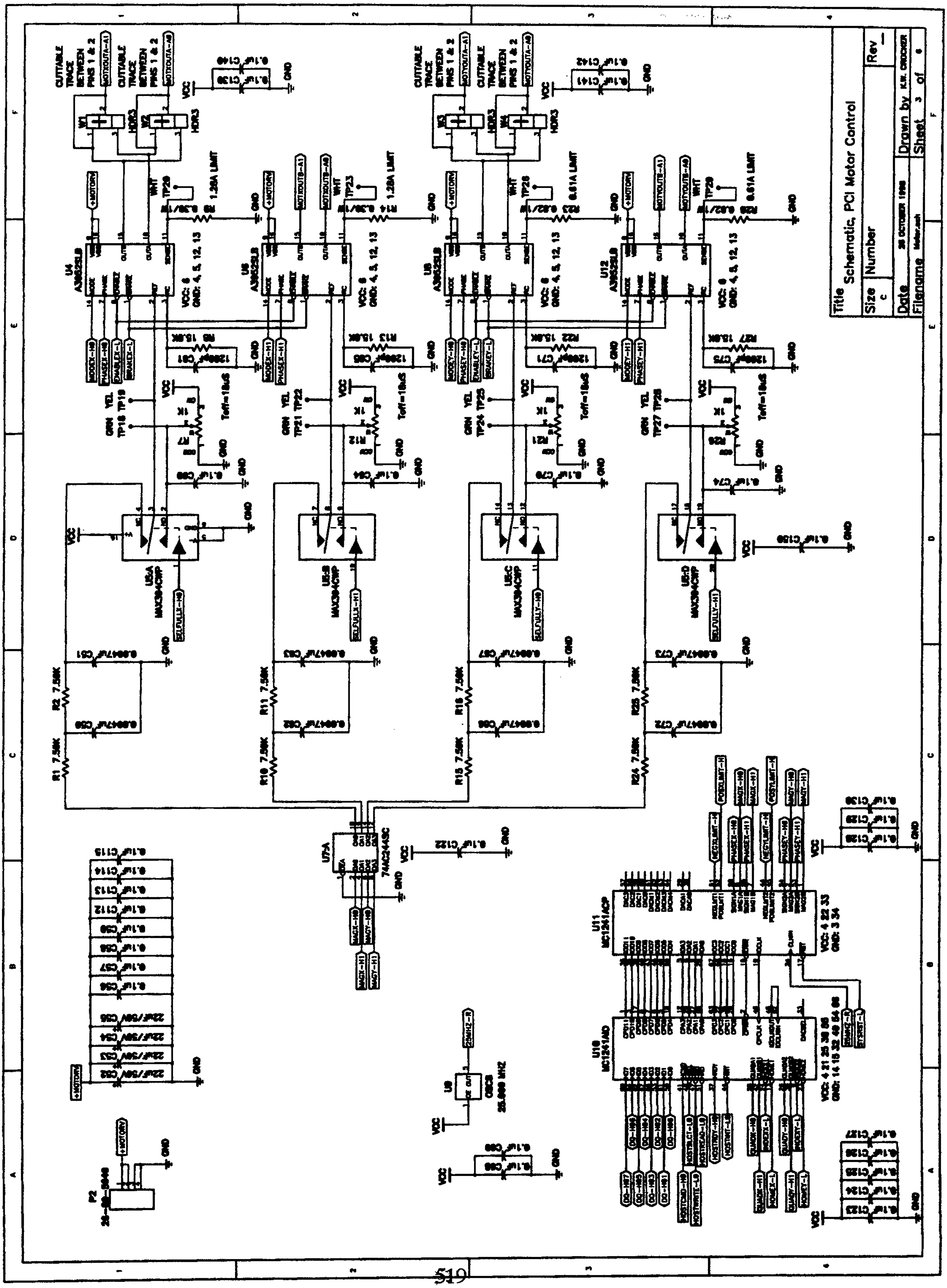
Date 27 OCTOBER 1998	Drawn by K.W. CROCKER
Filename Link_c.sch	Sheet 3 of 3

KENDAL CORP. TRADE SECRETS

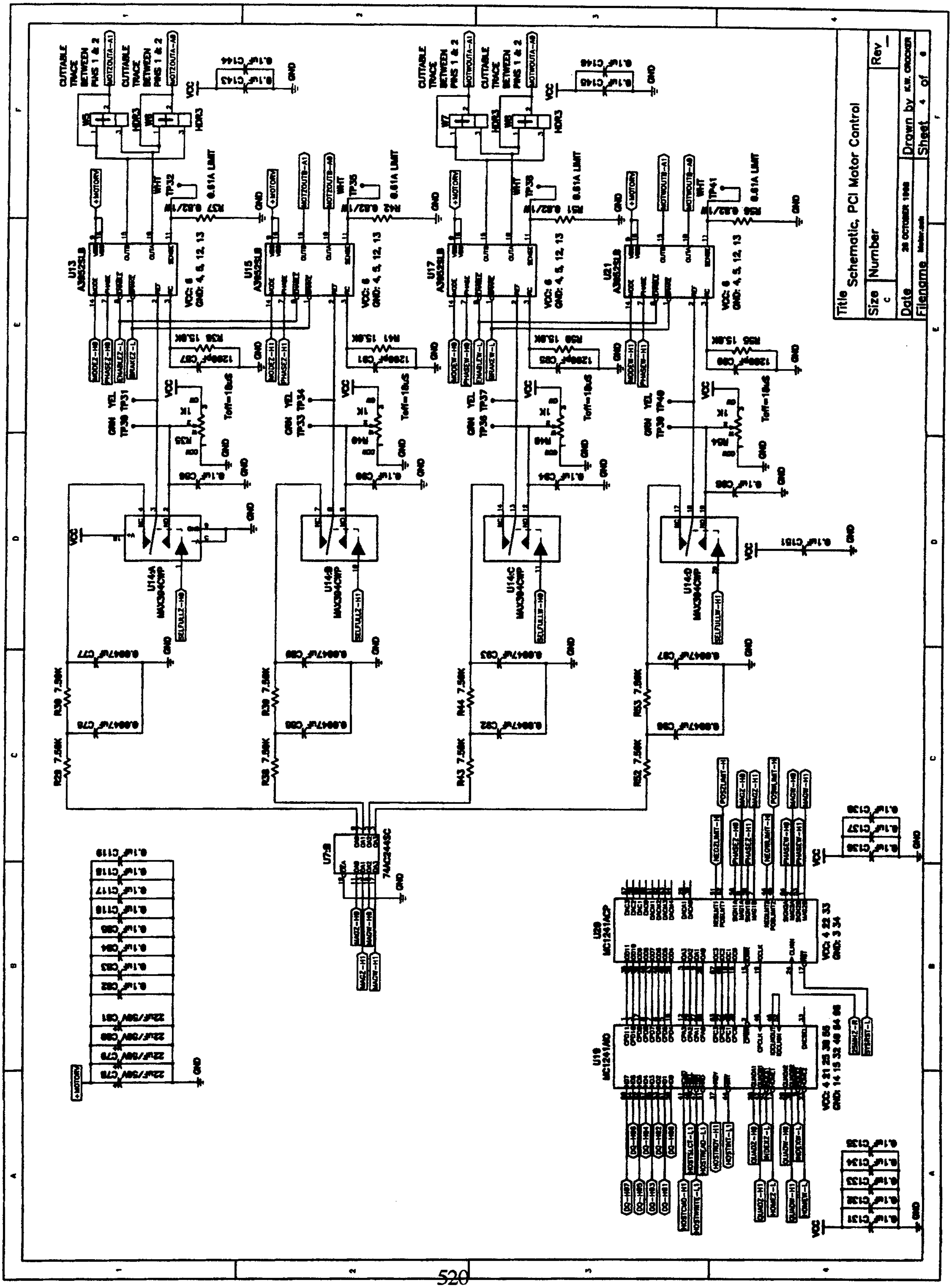


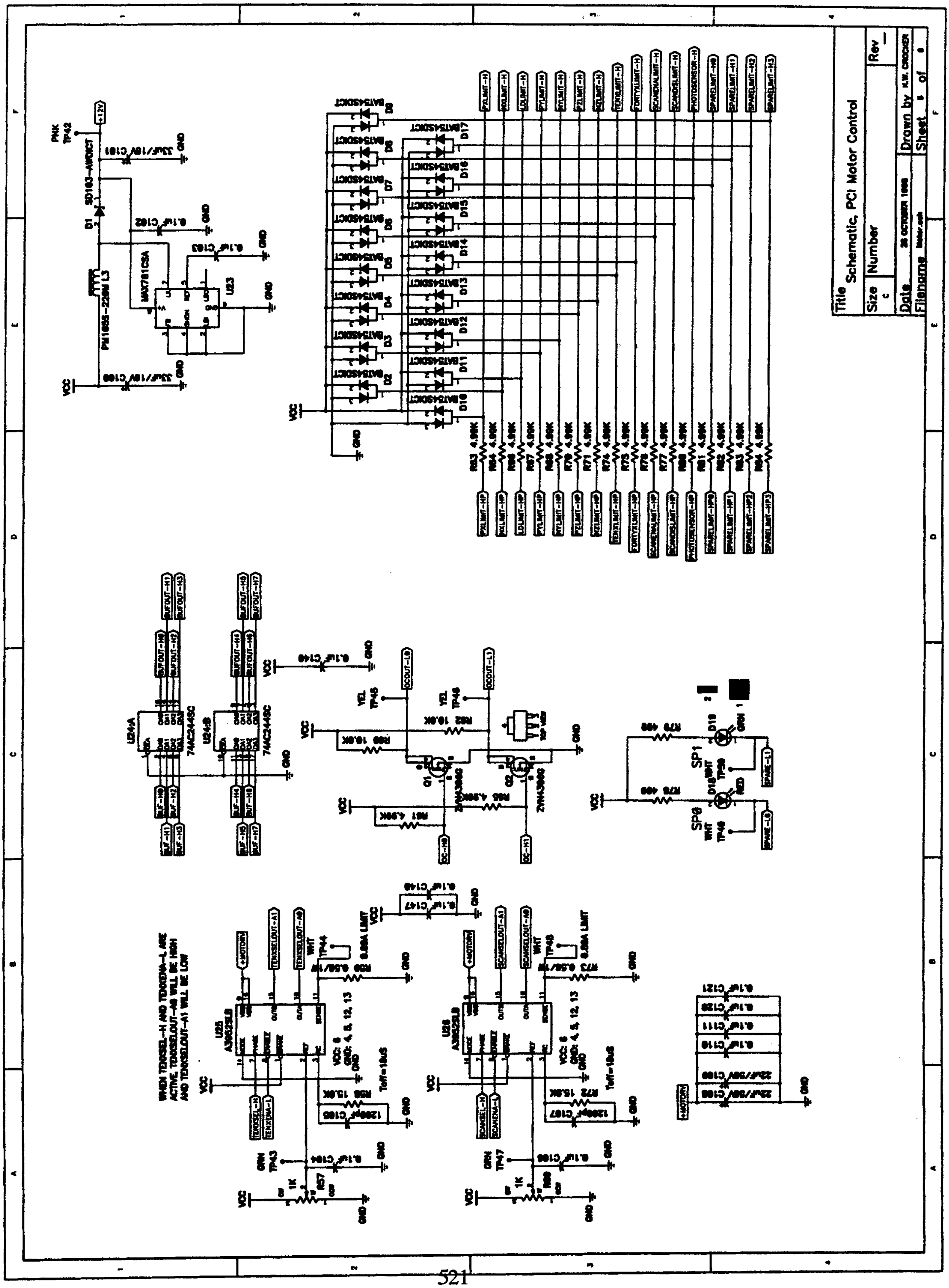

Title Schematic, PCI Motor Control
Size C
Number
Rev
Date
Drawn by
Filename
Sheet of
In System Programming (ISP) Header Connector
J1
CONN10M
VCC
GND
TCK-H
TMS-H
TDI-H
TDO-H
U3

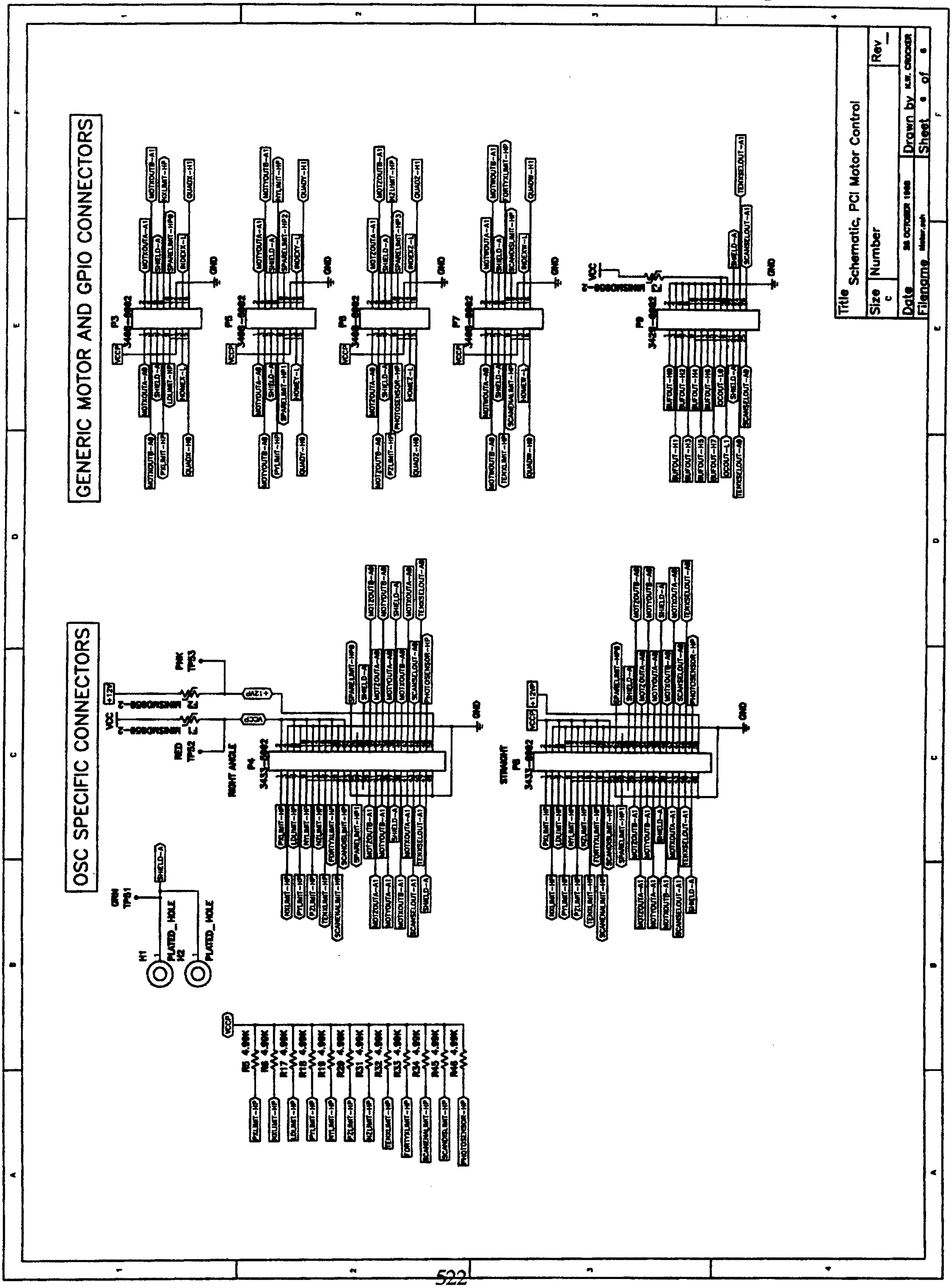
KENSAL CORP. TRADE SECRETS



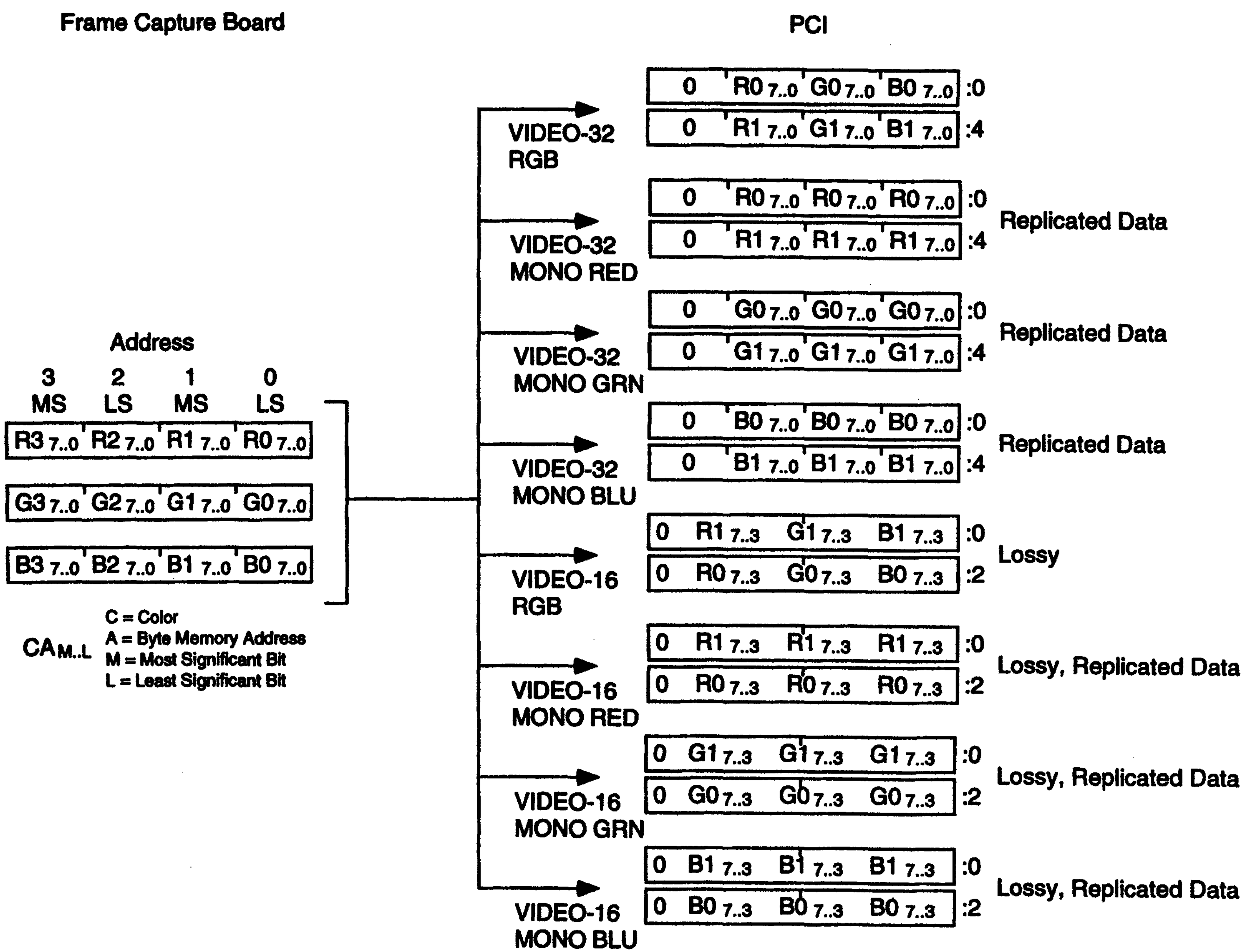
KENSAL CORP. TRADE SECRETS



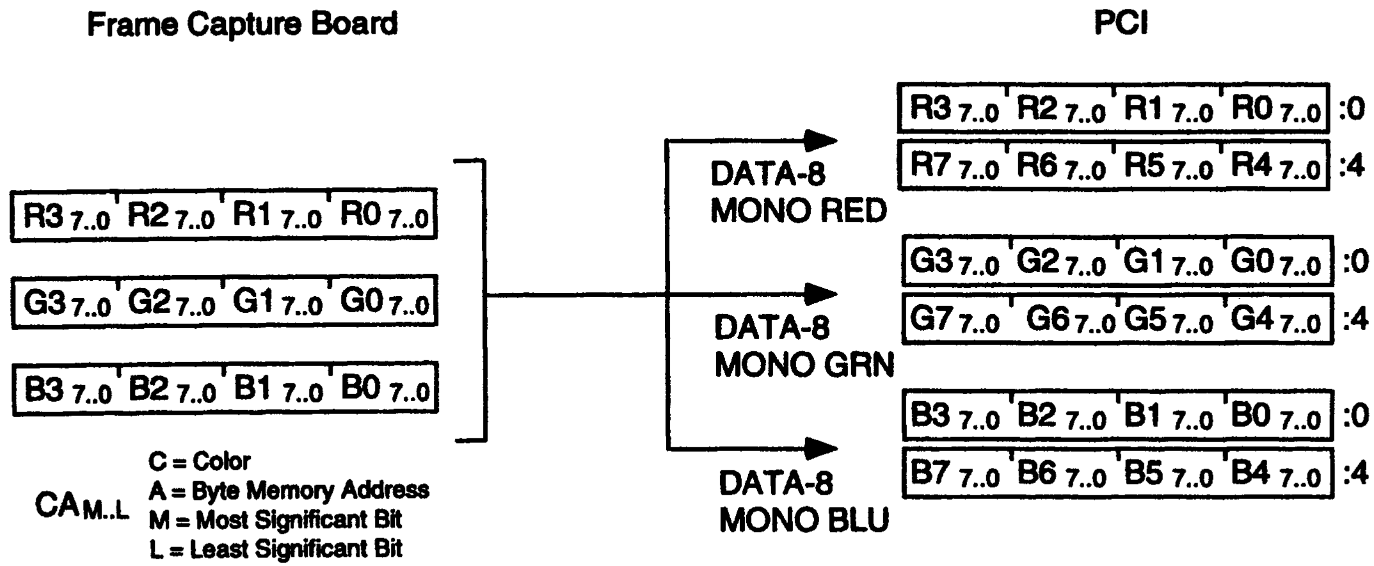

Title
Schematic, PCI Motor Control
Size
C
Number
Rev
Date
Drawn by
Filename
Sheet
of

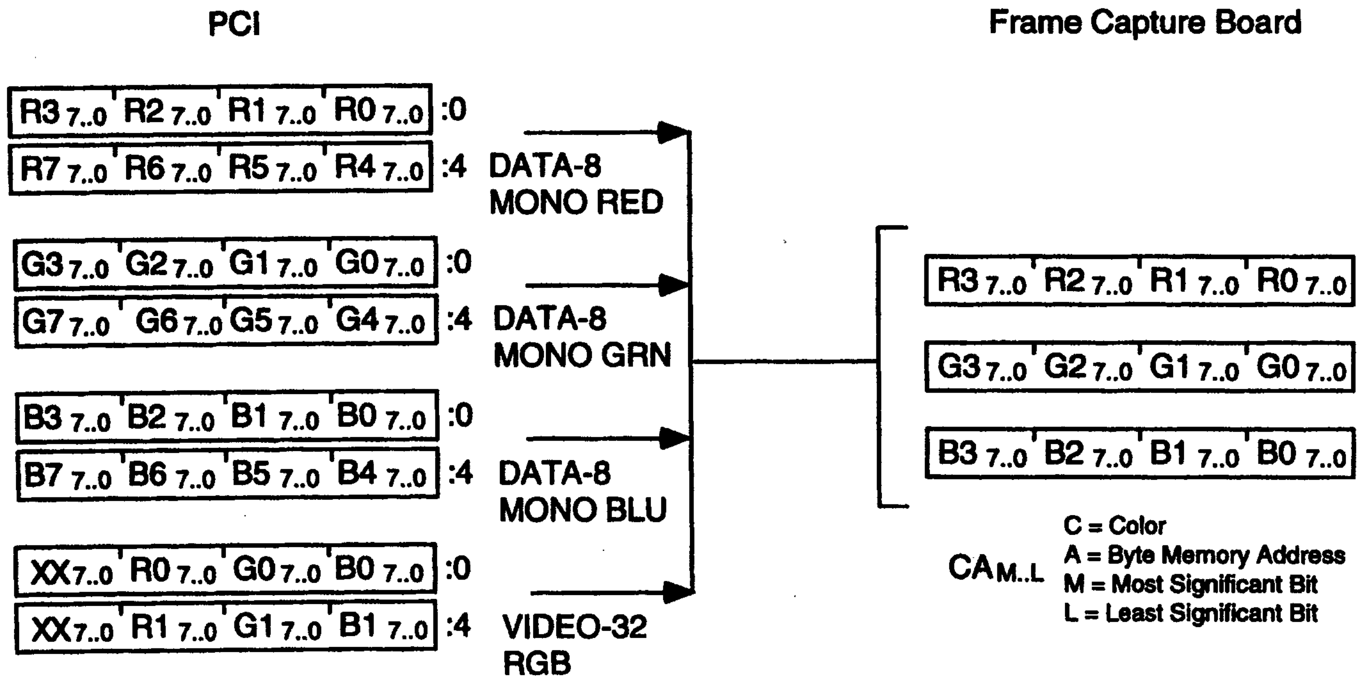

GENERIC MOTOR AND GPIO CONNECTORS
OSC SPECIFIC CONNECTORS
Title
Schematic, PCI Motor Control
Size
C
Number
Rev
Date
Drawn by
Filename
Sheet
of

KENSAL CORP. TRADE SECRETS



KENSAL CORP. TRADE SECRETS





TPW Database

Table/Column	Data Type	Description
Images Table:		The images table is designed to hold all guide image specific information. The images table can be seen as the primary table in the database.
image_id	Serial	images table primary key.
device_id	Integer	foreign key to the devices table.
institute_id	Integer	foreign key to the institutes table.
md_id	Integer	foreign key to the MD table.
operator_id	Integer	foreign key to the operators table.
folder	String	folder name of the image and it's associated files (8 characters or less ISO9660).
slide_no	String	7 digit medical slide number.
file_name	String	file name of the image file.
prefix	String	a slide number prefix.
suffix	String	a slide number suffix.
stain	String	a code indicating the stain used on the slide.
start_dt	Date	the date which the guide image was recorded or the case was started.
finish_dt	Date	the date which the case was finished.
topography	String	SNOMED topography code.
morphology	String	SNOMED morphology code.
function	String	SNOMED function code.
etiology	String	SNOMED etiology code.
living_org	String	SNOMED etiology: living organism code.
chem_etc	String	SNOMED etiology: chemicals, etc. code.
phys_agents	String	SNOMED etiology: physical agents code.
occupation	String	SNOMED occupation code.
social	String	SNOMED social context code.
disease	String	SNOMED disease code.
procedure	String	SNOMED procedure code.
general	String	SNOMED general code.
description	String	full text description of the image.
scale	Real	the scale of the image in microns per pixel.
xoffset	Integer	the offset from the label of the slide to the center of the image in microns (slide label held in right hand).
yoffset	Integer	the offset from the top of the slide to the center of the image in microns (slide label held in right hand).
thumbnail	Blob	thumbnail stored as a flattened PixMap (256x128).
results_name	String	file name of the results file.
Devices Table:		All of the individual devices which are recognized by the system will have an entry in the devices table. This makes it quite simple in the future to implement such features and CD-ROM burning utilities.
device_id	Serial	devices table primary key.
media_type	Integer	a 4 character mnemonic indicating the type of the media ('cdrm', 'locl', 'inet', ...).

name	String	the short name of the device.
path	String	the full path of the device.
create_dt	Integer	the date of device as applicable.
icon	Blob	an icon representing the device. Stored as a CIcon.
Institutes Table:		General information about the medical institution which originates the slides. This is not meant to be a comprehensive detailing of the institute.
institute_id	Serial	institutes table primary key.
name	String	institutes full name.
address	String	1st address line.
address_opt	String	2nd address line (suite no., bldg. no., etc.).
city	String	full city name.
state	String	2 character state abbreviation.
zip	String	postal code.
country	String	full country name (if blank USA assumed).
MD Table:		Information about the professionals who are primarily responsible for the diagnosis of the slides.
md_id	Serial	MD's table primary key.
fname	String	given name of MD
mname	String	initials of MD
lname	String	surname of MD
salutation	String	appropriate leading salutation (Mr., Mrs., Ms., etc.).
position	String	applicable position (Physician, Resident, etc.).
title	String	applicable title (President, Chief Resident, etc.).
credentials	String	credentials and entitlements (MD, D.D.E., etc.).
address	String	1st address line.
address_opt	String	2nd address line (suite no., bldg. no., etc.).
city	String	full city name.
state	String	2 character state abbreviation.
zip	String	postal code.
country	String	full country name (if blank USA assumed).
Operators Table:		Information about the individuals who are responsible for image recording.
operator_id	Serial	operator tables primary key.
fname	String	given name of operator.
mname	String	initials of operator.
lname	String	surname of operator.
HighMags Table:		A table for the storage of the high magnification image logistical data.
highmag_id	Serial	highmag tables primary key.
image_id (1)	Integer	foreign key to images table.
file_name	String	file name of the image file.
sequence (1)	Integer	highmag image sequence within the guide images set of highmags.

record_dt	Date	date of recording the image.
record_tm	Time	time of recording the image.
magnification	Integer	ocular magnification of recording.
scale	Real	the scale of the image in microns per pixel.
xposition	Integer	horizontal position of the center point of the image with respect to the guide image in microns.
yposition	Integer	vertical position of the center point of the image with respect to the guide image in microns.
width	Integer	height of the image in pixels.
height	Integer	width of the image in pixels.
operator_id	Integer	foreign key to the operators table (the operator who recorded the image).
1 - A composite index on image_id and sequence is used to ensure uniqueness.		

Kensal Foundation Class:		
KFCBlockSmooth	NA	An object for performing JPEG block smoothing
KFCButton	CButton	A multi-purpose button which allows different special effects.
KFCColor	KFCServices	An abstract color services classes which is functionally derived by several service objects.
KFCColorGrayRGBtoYCC	KFCColorRGBtoYCC	A service object for Grayscale RGB to YCbCr colorspace conversion.
KFCColorGStoYCC	KFCColorYCCtoYCC	A service object for Grayscale to YCbCr colorspace conversion.
KFCColorRGBtoYCC	KFCColor	A service object for RGB to YCbCr colorspace conversion.
KFCColorYCCtoGS	KFCColorYCCtoYCC	A service object for YCbCr to Grayscale colorspace conversion.
KFCColorYCCtoRGB	KFCColor	A service object for YCbCr to RGB colorspace conversion.
KFCColorYCCtoYCC	KFCColor	A service object for No colorspace conversion.
KFCCommFTP	KFCCommInternet	A communication object for accessing the internet via FTP.
KFCCommHTTP	KFCCommInternet	A communication object for accessing the internet via HTTP.
KFCCommInternet	KFCCommunication	A communication object for accessing the internet.
KFCCommunication	N/A	The virtual communication class.
KFCCompressor	N/A	A class for compressing some space.
KFCCompressorJPEG	KFCCompressor	JPEG compression object.
KFCCompressorJPEGList	CVoidPtrArray	A compressor list.
KFCCompSettings	N/A	Compressor settings.
KFCCompSettingsJPEG	KFCCompSettings	JPEG compressor settings.
KFCCompSettingsJPEGDicom	KFCCompSettingsJPEG	DICOM JPEG compressor settings.
KFCConfirm3StateDlg	CDialogDirector	A dialog box which will return one of three possible results (-1, 0, 1).
KFCConfirmDlg	CDialogDirector	A dialog box which will return one of two possible results (0, 1).
KFCConfirmListDlg	CDialogDirector	A dialog box which takes a pointer to a KFCStringList object and allows the user to select an item. The dialog returns the item number selected or 0 if cancelled.
KFCDebug	NA	A file containing helpful debugging routines (not an object).
KFCDesktop	CDesktop	A desktop object for floating window click sensing and menu bar hiding.
KFCDrawText.cp	CStaticText	Draws the text in an a transparent mode.
KFCEditText.cp	CEditText	Allows for the trapping of Fkeys.
KFCErrorDlg	CDialogDirector	A dialog box for alerting the user of error conditions.
KFCExpanderBtn	CIconButton	An icon button which remembers the expansion value.
KFCFieldText	CDialogText	A text object for displaying static text in a field (box).

KFCFile	CDataFile	A generic file handler.
KFCFileDicom	KFCFileImage	The entry point for DICOM file management
KFCFileDicomDir	KFCFileDicomObject	An object for supporting the DICOMDIR file.
KFCFileDicomImage	KFCFileDicomObject	A DICOM image object
KFCFileDicomMetadata	KFCFileDicomTagList	A DICOM metadata object in a DICOM file
KFCFileDicomObject	NA	A generic DICOM object (List of Tags)
KFCFileDicomObjectList	CVoidPtrArray	A list of DICOM objects
KFCFileDicomTag	NA	A generic DICOM tag
KFCFileDicomTagImage	KFCFileDicomTag	A DICOM image tag
KFCFileDicomTagImplicit	KFCFileDicomTag	An implicit DICOM tag
KFCFileDicomTagList	CVoidPtrArray	A list of DICOM tags
KFCFileDicomTagSequence	KFCFileDicomTag	A DICOM tag sequence
KFCFileDicomTagSequenceIten	KFCFileDicomTagSequence	A DICOM tag sequence item
KFCFileGIF	KFCFileImage	A robust GIF file manager.
KFCFileGIFImage	N/A	A GIF Image in a GIF file.
KFCFileGIFImageList	CVoidPtrArray	A list of GIF Images derived as CPtrArray<KFCFileGIFImage>.
KFCFileImage	KFCFile	A generic image file handler.
KFCFileJPEG	KFCFileImage	A robust JPEG file manager.
KFCFileQuickTime	KFCFileSound	A sound file class for working with quicktime files.
KFCFileSound	KFCFile	A format free sound file class.
KFCFileTIFF	KFCFileImage	A robust TIFF file manager.
KFCFileTIFFImage	N/A	A TIFF Image in a TIFF file.
KFCFileTIFFImageList	CVoidPtrArray	A list of TIFF images within one file derived as CPtrArray<KFCFileTIFFTagList>.
KFCFileTIFFTag	N/A	A TIFF Tag member of the TIFF file.
KFCFileWave	KFCFileSound	A wave file management object
KFCFlexiblePICTGrid	CPICTGrid	A PICT grid with customizable selection mechanisms.
KFCGlobs	N/A	An object for providing some static functions and global variable initialization.
KFCGridScroll	CScrollPane	A scroll bar for scrolling according to a pre-defined grid dimension.
KFCHuffman	KFCServices	An abstract huffman object
KFCHuffmanDecode	KFCHuffman	An object for decoding huffman data
KFCHuffmanEncode	KFCHuffman	An object for encoding huffman data
KFCIdleChores	CChore	A class for portioning out time to tasks at idle time.
KFCImage	N/A	A virtual class for image management.
KFCImageDims	N/A	An object for storiing a standardized image description.
KFCImageGWorld	KFCImageHead	A GWorld object derived as KFCImageHead<GWorldPtr>.
KFCImageList	CVoidPtrArray	A list of images used by the KFCFlexiblePICTGrid.
KFCImagePane	CPanorama	An object which allows any kind of an image to be displayed and sense clicks and scroll.

KFCImagePICT	KFCImageHead	A PICT object derived as KFCImageHead<PicHandle>.
KFCImagePixMap	KFCImageHead	A PixMap object derived as KFCImageHead<PixMapPtr>.
KFCJPEGPipe	NA	A generic JPEG controller
KFCJPEGPipeComplex	KFCJPEGPipe	A JPEG controller for handling multiple pass coding
KFCJPEGPipeComplexEntropy	KFCJPEGPipeComplex	A JPEG controller for entropy multiple pass coding
KFCJPEGPipeEntropy	KFCJPEGPipe	A JPEG controller of entropy single pass coding
KFCMCU	KFCServices	An abstract object for performing Discrete Cosine Transforms.
KFCMCUExtract	KFCMCU	A DCT Extraction and quantization object.
KFCMCUInsert	KFCMCU	A DCT disassembler object.
KFCMCUInsertInterleaved	KFCMCUInsert	An interleaved DCT disassembler.
KFCNetwork	N/A	An object which connections to a network.
KFCPasswordText	CDialogText	A text object for entering passwords (the input characters are masked).
KFCProgressBar	CRectOvalButton	A progress bar object.
KFCPtrArray	CPtrArray	An array template to provide a few more features than CPtrArray.
KFCQuantizer	KFCServices	An abstract object for performing color quantization services.
KFCQuantizer1Pass	KFCQuantizer	A single pass quantizer.
KFCQuantizer1Pass3Color	KFCQuantizer1Pass	A single pass quantizer for RGB images
KFCQuantizer1PassDither	KFCQuantizer1Pass	A single pass quantizer for dithered images
KFCQuantizer2Pass	KFCQuantizer	A double pass quantizer.
KFCRequestDlg	CDialogDirector	For requesting information from the user.
KFCSample	KFCServices	An abstract color sampling service object.
KFCSampleDn	KFCSample	A down sampling object
KFCSampleDnFull	KFCSampleDn	A full down sampling object
KFCSampleDnFullSmooth	KFCSampleDnFull	A full down sampling object that performs smoothing.
KFCSampleDnH2V1	KFCSampleDn	A 2:1 horizontal and 1:1 vertical down sampling object.
KFCSampleDnH2V2	KFCSampleDn	A 2:1 horizontal and 2:1 vertical down sampling object.
KFCSampleDnH2V2Smooth	KFCSampleDnH2V2	A 2:1 horizontal and 2:1 vertical down sampling object with smoothing.
KFCSampleDnInt	KFCSampleDn	An arbitrary integral down sampling object.
KFCSampleUp	KFCSample	An up sampling object
KFCSampleUpFull	KFCSampleUp	A full up sampling object
KFCSampleUpH2V1	KFCSampleUp	A 2:1 horizontal and 1:1 vertical up sampling object.
KFCSampleUpH2V2	KFCSampleUp	A 2:1 horizontal and 2:1 vertical up sampling object.
KFCSampleUpInt	KFCSampleUp	An arbitrary integral up sampling object.
KFCSlider	CSubViewDisplayer	A generic slider control.
KFCSliderBar	CPictureButton	The buttons in the slider.

KFCSliderBtn	CIconButton	The bar in the slider.
KFCSliderTE	CDialogText	The active text box associated with a slider bar.
KFCSliderTX	CStaticText	The inactive text box associated with a slider bar.
KFCSliderThumb	CIconButton	The thumb in the slider.
KFCStream.cp	CFileStream	A stream object with a configuable buffer.
KFCString.cp	N/A	A string object which is inheritable.
KFCStringArrayPane.cp	CArrayPane	An array pane of strings.
KFCStringList.cp	CVoidPtrArray	An array of strings.
KFCTask	N/A	A virtual task class.
KFCTaskList	CVoidPtrArray	A list of tasks.
KFCTaskProg	KFCTask	A generic progress task. This class does not do any drawing.
KFCTaskRdComm	KFCTask	A communication task class for reading from a remote location into a local file.
KFCTaskSound	KFCTask	A task object for playing sound objects.
KFCUtilsFile	N/A	Some generic and useful file functions.
KFCUtilsGen	N/A	Some generic and useful static functions.
KFCUtilsString	N/A	An object for providing some static string functions.
KFCVersion	N/A	A version control object.
Tele-Pathology Workstation Classes:		
TPWAddressArrayPane	CArrayPane	An object for displaying the addresses list.
TPWAddresses	CVoidPtrArray	An object for holding an array of address entries.
TPWAddressEntry	NA	An individual address.
TPWApp	CApplication	Handle all command parsing and switching. Perform global instance management.
TPWCamera	N/A	The primary object for manipulating the Frame and Motor drivers.
TPWCameraLUT	N/A	An object for reading and writing LUT
TPWConnectionsDlg	CDialogDirector	Used to enter connections information for the valid addresses in the transmit dialog.
TPWDatabase	N/A	A Database interface object.
TPWDatabaseQuery	N/A	The object which remembers and manages the users queries on the database.
TPWDatabaseSchema	N/A	An object which is responsible for the creation of the entire database structure.
TPWDBEnterDlg	CDialogDirector	For gathering image data on diagnosis.
TPWDBImagesDlg	CDialogDirector	Handles all file retrieval for viewing of saved images. Is also responsible for displaying thumbnail images on screen.
TPWDBSearchDlg	CDialogDirector	For entering database search criteria.
TPWFileDiagnosis	KFCFileText	A file object for diagnoses
TPWFileMessages	KFCFileText	A file object for messages
TPWFilePrefs	KFCFileText	A file object for preferences
TPWFileRegions	KFCFileText	A file object for regions
TPWFKeyEdit	CDialogDirector	A dialog object which allows the user to costomize which Fkeys are used during voice transcription.

TPWFocus	CDialogDirector	The focus control for the TPW camera.
TPWFrameDriverDlg	CDialogDirector	The dialog for low level control of the frame and motor drivers.
TPWFrameFWind	CFloatDirector	A floating window version of the Frame Driver Dialog.
TPWFrameLogArrayPane	CArrayPane	An array pane for displaying the Frame and Motor driver logs.
TPWFrameLUTArrayPane	CArrayPane	An array pane for displaying Frame driver LUT values.
TPWGuideSV		Displays the current region of interest during high magnification image creation at 20x.
TPWHelpDlg	CDialogDirector	The help dialog for the TPW.
TPWImageDataFWind	CFloatDirector	A display of the image data in the current record.
TPWImagePort	KFCImagePane	The view port will handle displaying of the images as well as sensing mouse clicks and performing pan and scroll with the slide table.
TPWLoginDlg	CDialogDirector	The main login dialog at startup in a multiple user environment.
TPWLUTSlider	KFCSlider	A slide control for adjusting LUT values.
TPWMagGuide	CSubViewDisplayer	Handles all software magnifications of the guide image.
TPWMagVideo	CSubViewDisplayer	Performs all hardware high magnification magnifications as well as indicating the current magnification factor. Also indicates and receives the users location magnification request.
TPWMainWind	CDocument	All functions are initiated and handled through the Main Window. This object is responsible for setting the screen objects and switching to the appropriate methods for actions chosen by the user.
TPWMessage	N/A	A message object
TPWMessagesArrayPane	CArrayPane	An array pane for displaying messages
TPWMessageList	CVoidPtrArray	An array of messages
TPWMessagesDlg	CDialogDirector	Remote Users Messages Dialog.
TPWMotorScript	N/A	Handles script parsing for running motor scripts in the Frame Driver Dialog: Motor controller.
TPWOverlay	CSubViewDisplayer	To change the overlay display.
TPWOverlayData	N/A	An object used to describe an individual high magnification image region.
TPWOverlayList	CSubViewDisplayer	A List of TPWOverlayData Objects.
TPWPreferencesDlg	CDialogDirector	Preferences setting dialog.
TPWProgress	CSubViewDisplayer	The progress control object.
TPWRegion	N/A	A region class.
TPWRegionList	CVoidPtrArray	A list of regions.
TPWSlideInfo	N/A	A slide information class.
TPWSlideNoDlg	CDialogDirector	A dialog to capture the slide information from the user.
TPWSplash	CDialogDirector	The Splash screen which is displayed on screen briefly during launch.

TPWTaskCamera	KFCTask	An abstract class to handle frame and motor driver asynchronous functionality.
TPWTaskCameraLive	TPWTaskCamera	The task responsible for monitoring and maintaing live video.
TPWTaskCameraPict	TPWTaskCamera	A class to get a picture from the camera.
TPWTaskCameraScan	TPWTaskCamera	Manages the progress and control during a scanning operation.
TPWTaskCoordinator	KFCTask	The initiating task to perform file transfers.
TPWTaskFrame	KFCTask	A task class to manage various ongoing tasks while the frame and motor driver dialog is in use.
TPWTaskListener	KFCTask	A class for listening for incoming communication.
TPWTaskMakeThumbnail	KFCTask	For building a thumbnail in the background.
TPWTaskParticipant	KFCTask	The subserviant class for performing file transfers.
TPWTaskProg	KFCTaskProg	The progress bar drawing class.
TPWTaskRdImage	KFCTask	The image file reading class which draws to the image port.
TPWTaskVoiceDlg	KFCTask	An object for updating progress information and button status during voice play.
TPWTaskWrHighMag	TPWTaskWrImage	A task which captures and then writes a High Mag image.
TPWTaskWrImage	KFCTask	A task for performing the write functions of an image.
TPWThumbnail	KFCImageGWorld	The object responsible for managing a thumbnail in the guide mosaic.
TPWTransmitDlg	CDialogDirector	Gathers the recipients address and sends the request to TPWCommunication.
TPWUserPrefList	N/A	A class to store user level preferences.
TPWVoiceDlg	CDialogDirector	Performs all sound manipulation tasks. Is also responsible for attaining diagnosis transcription.
RC/21 Objects:		
BADObject	RCObject	
BRANCH	N/A	
BTree	N/A	
BTREE_CURSOR	N/A	
B_POSITION	N/A	
Column	RCObject	
Database	RCObject	
DBVALUE	N/A	
DCE	N/A	
Dictionary	N/A	
DTE	N/A	
Fieldmap	N/A	
FIELDPAGE	N/A	
FILTER	N/A	
FilterSearchObject	SearchObject	
IMPT	RCObject	
IMPTE	N/A	

losTie	N/A	
losUnitbuf	N/A	
NAME_ENTRY	N/A	
NECESSARY_RELATION	N/A	
RCObject	N/A	
SearchObject	RCObject	
streamoff	N/A	
streampos	N/A	
Table	RCObject	
TASK	N/A	
TOKEN	N/A	
VALDESC	N/A	
VALUE	N/A	
RC/21 Add-in Classes:		
RCColumnBLOB	Column	An object for implementing the movement of PixMap data into and out of the BLOB column.
RCSerialDatabase	Database	For implementation of the serialized database.
RCSerialTable	Table	For implementation of the serialized table.
TPW Database Importer Classes:		
TPDApp	CApplication	The importer application class.
TPDImportInternet	N/A	The object responsible for importing internet information.
TPDImportLocal	N/A	The object responsible for importing local information.
TPDImportLocalDCM	TPDImportLocal	The object responsible for importing local DICOM information.
TPDiNetPrefs	CDialogDirector	The Internet importing preferences dialog.
TPDMain	CSaver	An object to provide a main interface derived from CSaver<CCollaborator>.

Recd 1/11/2000

DEPARTMENT OF THE ARMY
US ARMY MEDICAL RESEARCH AND MATERIEL COMMAND
504 SCOTT STREET
FORT DETRICK, MARYLAND 21702-5012

REPLY TO
ATTENTION OF:

MCMR-RMI-S (70-1y) 4 Jan 00

MEMORANDUM FOR Administrator, Defense Technical Information Center, ATTN: DTIC-OCA, 8725 John J. Kingman Road, Fort Belvoir, VA 22060-6218

SUBJECT: Request Change in Distribution Statement

1. The U.S. Army Medical Research and Materiel Command has reexamined the need for the limitation assigned to technical reports written for the attached Grants. Request the limited distribution statements for Accession Document Numbers listed be changed to "Approved for public release; distribution unlimited." This report should be released to the National Technical Information Service.

2. Point of contact for this request is Ms. Judy Pawlus at DSN 343-7322 or by email at Judy.Pawlus@amedd.army.mil.

FOR THE COMMANDER:

Phylis Rinehart
PHYLIS M. RINEHART
Deputy Chief of Staff for
Information Management

94-J-4500 AD-B244332

Completed 1-13-00